Open Source Used In AppDynamics_GoToMarketService 23.1

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.

Text Part Number: 78EE117C99-1519128524

This document contains licenses and notices for open source software used in this product. With respect to the free/open source software listed in this document, if you have any questions or wish to receive a copy of any source code to which you may be entitled under the applicable free/open source license(s) (such as the GNU Lesser/General Public License), please submit this form.

In your requests please include the following reference number 78EE117C99-1519128524

Contents

1.1 java-architecture-for-xml-binding 3.0.0
1.1.1 Available under license
1.2 opentelemetry-java---io.opentelemetry:opentelemetry-sdk-metrics 1.14.0
1.2.1 Available under license
1.3 spring-cloud-sleuth-instrumentation 3.1.3
1.3.1 Available under license
1.4 jul-to-slf4j-bridge 1.7.36
1.4.1 Available under license
1.5 brave-instrumentation-spring-rabbitmq 5.13.9
1.5.1 Available under license
1.6 testcontainers-jdbc 1.17.1
1.6.1 Available under license
1.7 json-library-from-android-sdk 0.0.20131108.vaadin1
1.7.1 Available under license
1.8 snake-yaml 1.30
1.8.1 Available under license
1.9 testcontainers-jdbc-postgresql 1.17.1
1.9.1 Available under license
1.10 brave-instrumentation-apache-httpasyncclient-v 5.13.9
1.10.1 Available under license
1.11 zjsonpatch 0.3.0
1.11.1 Available under license
1.12 jackson-core 2.13.3
1.12.1 Available under license
1.13 fabric8-::-kubernetes-model-::-core 5.12.1
1.13.1 Available under license

1.14 netty-project 4.1.79.Final
1.14.1 Available under license
1.15 jackson-datatype-jsr310 2.13.3
1.15.1 Available under license
1.16 xmlunit-xmlunit-core 2.9.0
1.16.1 Available under license
1.17 fabric8-::-kubernetes-model-::-node 5.12.1
1.17.1 Available under license
1.18 apache-log4j-jul-adapter 2.17.2
1.18.1 Available under license
1.19 junit-platform-junit-platform-commons 1.8.2
1.19.1 Available under license
1.20 fabric8-::-kubernetes-model-::-extensions 5.12.1
1.20.1 Available under license
1.21 netty/transport/classes/epoll 4.1.79.Final
1.21.1 Available under license
1.22 java-architecture-for-xml-binding 2.3.1
1.22.1 Available under license
1.23 spring-boot-actuator 2.7.2
1.23.1 Available under license
1.24 fabric8-::-kubernetes-model-::-common 5.12.1
1.24.1 Available under license
1.25 fabric8-::-kubernetes-model-::-admission-registration,-authentication-and-authorization 5.12.1
1.25.1 Available under license
1.26 spring-webflux 5.3.22
1.26.1 Available under license
1.27 spring-cloud-starter-sleuth 3.1.3
1.27.1 Available under license
1.28 asm-tree 9.1
1.28.1 Available under license
1.29 postgresql-jdbc-driver-pgjdbc 42.3.3
1.29.1 Available under license
1.30 spring-boot 2.7.2
1.30.1 Available under license
1.31 javassist 3_26_0_ga
1.31.1 Available under license
1.32 fabric8-::-kubernetes-model-::-certificates 5.12.1
1.32.1 Available under license

1.33 error_prone_annotations 2.7.1
1.33.1 Available under license
1.34 aspectj-runtime 1.9.7
1.34.1 Available under license
1.35 spring-boot-reactor-netty-starter 2.7.2
1.35.1 Available under license
1.36 mac-os 4.1.80.Final
1.36.1 Available under license
1.37 testcontainers-core 1.17.1
1.37.1 Available under license
1.38 zipkin-v2 2.23.2
1.38.1 Available under license
1.39 json-path 2.7.0
1.39.1 Available under license
1.40 hibernate-validator 6.2.3.Final
1.40.1 Available under license
1.41 apache-commons-collections 3.2.2
1.41.1 Available under license
1.42 okhttp 4.9.3
1.42.1 Available under license
1.43 checker-qual 3.12.0
1.43.1 Available under license
1.44 fabric8-::-kubernetes-model-::-flowcontrol 5.12.1
1.44.1 Available under license
1.45 fabric8-::-kubernetes-model-::-networking 5.12.1
1.45.1 Available under license
1.46 docker-java-transport 3.2.13
1.46.1 Available under license
1.47 byte-buddy-agent 1.12.12
1.47.1 Available under license
1.48 asm 9.1
1.48.1 Available under license
1.49 j2objc-annotations 1.3
1.49.1 Available under license
1.50 spring-boot-log4j2-starter 2.7.2
1.50.1 Available under license
1.51 jackson-annotations 2.13.3
1.51.1 Available under license
1.52 spring-boot-test-auto-configure 2.7.2

1.52.1 Available under license
1.53 brave-instrumentation-kafka-streams 5.13.9
1.53.1 Available under license
1.54 jooq-meta-extensions 3.16.4
1.54.1 Available under license
1.55 junit-jupiter-junit-jupiter-api 5.8.2
1.55.1 Available under license
1.56 spring-retry 1.3.1
1.56.1 Available under license
1.57 jooq-meta 3.16.4
1.57.1 Available under license
1.58 apache-commons-lang 3.12.0
1.58.1 Available under license
1.59 jakarta-activation 1.2.2
1.59.1 Available under license
1.60 fabric8-::-kubernetes-model-::-scheduling 5.12.1
1.60.1 Available under license
1.61 jackson-datatype-jdk8 2.13.3
1.61.1 Available under license
1.62 asm-tree 9.2
1.62.1 Available under license
1.63 fabric8-::-kubernetes-model-::-autoscaling 5.12.1
1.63.1 Available under license
1.64 spring-cloud-sleuth-api 3.1.3
1.64.1 Available under license
1.65 bouncy-castle 1.68
1.65.1 Available under license
1.66 awaitility 4.2.0
1.66.1 Available under license
1.67 assertj-fluent-assertions 3.22.0
1.67.1 Available under license
1.68 commons-compress 1.21
1.68.1 Available under license
1.69 saxon-xslt-and-xquery-processor 10.6
1.69.1 Available under license
1.70 brave-instrumentation-http-adapters 5.13.9
1.70.1 Available under license
1.71 hikaricp HikariCP-4.0.3
1.71.1 Available under license

1.72 brave-context:-slf4j 5.13.9
1.72.1 Available under license
1.73 zipkin-reporter-metrics-by-micrometer 2.16.3
1.73.1 Available under license
1.74 spring-cloud-commons 3.1.3
1.74.1 Available under license
1.75 okhttp-logging-interceptor 4.9.3
1.75.1 Available under license
1.76 reflections 0.9.12
1.76.1 Available under license
1.77 mockwebserver 4.10.0
1.77.1 Available under license
1.78 fabric8-::-kubernetes-model-::-rbac 5.12.1
1.78.1 Available under license
1.79 brave-instrumentation-jms 5.13.9
1.79.1 Available under license
1.80 jna 5.8.0
1.80.1 Available under license
1.81 apiguardian-apiguardian-api 1.1.2
1.81.1 Available under license
1.82 asm-commons 9.1
1.82.1 Available under license
1.83 tomcat 9.0.65
1.83.1 Available under license
1.84 slf4j-api-module 1.7.36
1.84.1 Available under license
1.85 zipkin-reporter-core 2.16.3
1.85.1 Available under license
1.86 generex 1.0.2
1.86.1 Available under license
1.87 jetbrains-annotations 13.0
1.87.1 Available under license
1.88 fabric8-::-kubernetes-model-::-batch 5.12.1
1.88.1 Available under license
1.89 micrometer-metrics/micrometer 1.9.2
1.89.1 Available under license
1.90 spring-boot-test 2.7.2
1.90.1 Available under license
1.91 hdrhistogram 2.1.12

1.91.1 Available under license
1.92 kotlin 1.6.21
1.92.1 Available under license
1.93 guava-internalfuturefailureaccess-and-internalfutures 1.0.1
1.93.1 Available under license
1.94 automation 1.11-8
1.94.1 Available under license
1.95 guava 31.0.1-jre
1.95.1 Available under license
1.96 junit-5-bill-of-materials 5.8.2
1.96.1 Available under license
1.97 fabric8-::-kubernetes-model-::-api-extensions 5.12.1
1.97.1 Available under license
1.98 spring-cloud-sleuth-autoconfigure 3.1.3
1.98.1 Available under license
1.99 asm-based-accessors-helper-used-by-json-smart 2.4.8
1.99.1 Available under license
1.100 apache-log4j 2.17.2
1.100.1 Available under license
1.101 brave-instrumentation-kafka-clients 5.13.9
1.101.1 Available under license
1.102 project-lombok 1.18.24
1.102.1 Available under license
1.103 junit-jupiter-junit-jupiter-params 5.8.2
1.103.1 Available under license
1.104 jsonassert 1.5.1
1.104.1 Available under license
1.105 testcontainers-database-commons 1.17.1
1.105.1 Available under license
1.106 reactive-relational-database-connectivity-spi 0.9.1.RELEASE
1.106.1 Available under license
1.107 jooq 3.16.4
1.107.1 Available under license
1.108 opentelemetry-java 1.14.0-alpha
1.108.1 Available under license
1.109 brave-instrumentation-messaging-adapters 5.13.9
1.109.1 Available under license
1.110 brave-instrumentation-apache-httpclient-v 5.13.9
1.110.1 Available under license

1.111 spring-security-rsa 1.0.10.RELEASE
1.111.1 Available under license
1.112 fabric8-::-kubernetes-model-::-events 5.12.1
1.112.1 Available under license
1.113 spring-cloud-starter 3.1.3
1.113.1 Available under license
1.114 reactor-test-support 3.4.24
1.114.1 Available under license
1.115 lz4-and-xxhash 1.8.0
1.115.1 Available under license
1.116 jetbrains-kotlin-kotlin-stdlib-jdk7 1.6.21
1.116.1 Available under license
1.117 apache-log4j-slf4j-binding 2.17.2
1.117.1 Available under license
1.118 jacoco 0.8.7
1.118.1 Available under license
1.119 functionality-for-the-reactor-netty-library 1.0.21
1.119.1 Available under license
1.120 fabric8-::-kubernetes-model-::-storage-class 5.12.1
1.120.1 Available under license
1.121 mockito 4.5.1
1.121.1 Available under license
1.122 byte-buddy 1.12.12
1.122.1 Available under license
1.123 kotlin-stdlib-common 1.6.21
1.123.1 Available under license
1.124 jakarta-annotations-api 1.3.5
1.124.1 Available under license
1.125 guava-listenablefuture-only 9999.0-empty-to-avoid-conflict-with-guava
1.125.1 Available under license
1.126 docker-java 3.2.13
1.126.1 Available under license
1.127 jetbrains-kotlin-kotlin-stdlib-jdk8 1.6.21
1.127.1 Available under license
1.128 brave 5.13.9
1.128.1 Available under license
1.129 java-vault-driver 5.1.0
1.129.1 Available under license
1.130 okio 2.8.0

1.130.1 Available under license
1.131 okio 3.0.0
1.131.1 Available under license
1.132 spring-commons-logging-bridge 5.3.22
1.132.1 Available under license
1.133 duct-tape 1.0.8
1.133.1 Available under license
1.134 asm-analysis 9.2
1.134.1 Available under license
1.135 findbugs-jsr305 3.0.2
1.135.1 Available under license
1.136 apache-kafka 3.1.1
1.136.1 Available under license
1.137 jacoco 0.8.8
1.137.1 Available under license
1.138 picocli 4.6.1
1.138.1 Available under license
1.139 junit-jupiter-aggregator 5.8.2
1.139.1 Available under license
1.140 jackson-databind 2.13.3
1.140.1 Available under license
1.141 brave-propagation:-amazon-web-services-(aws) 0.21.3
1.141.1 Available under license
1.142 junit-platform-junit-platform-engine 1.8.2
1.142.1 Available under license
1.143 brave-instrumentation:-mongodb 5.13.9
1.143.1 Available under license
1.144 netty/resolver/dns/classes/macos 4.1.79.Final
1.144.1 Available under license
1.145 java-classmate classmate-1.5.1
1.145.1 Available under license
1.146 apache-log4j-layout-for-json-template 2.17.2
1.146.1 Available under license
1.147 spring-boot-actuator-autoconfigure 2.7.2
1.147.1 Available under license
1.148 spring-framework 5.3.22
1.148.1 Available under license
1.149 checker-qual 3.5.0
1.149.1 Available under license

1.150 docker-java-transport-zerodep 3.2.13
1.150.1 Available under license
1.151 jackson-module-parameter-names 2.13.3
1.151.1 Available under license
1.152 hamcrest v2.2
1.152.1 Available under license
1.153 open-telemetry/opentelemetry-java 1.14.0
1.153.1 Available under license
1.154 bean-validation-api 2.0.2
1.154.1 Available under license
1.155 fabric8-::-kubernetes-model-::-apps 5.12.1
1.155.1 Available under license
1.156 aspectj-weaver 1.9.7
1.156.1 Available under license
1.157 json-smart 2.4.8
1.157.1 Available under license
1.158 brave-instrumentation-rpc-adapters 5.13.9
1.158.1 Available under license
1.159 spring-boot-webflux-starter 2.7.2
1.159.1 Available under license
1.160 junit-jupiter-junit-jupiter-engine 5.8.2
1.160.1 Available under license
1.161 liquibase 4.9.1
1.161.1 Available under license
1.162 mac-os 4.1.79.Final
1.162.1 Available under license
1.163 mockito-junit-jupiter 4.5.1
1.163.1 Available under license
1.164 jetbrains-annotations 17.0.0
1.164.1 Available under license
1.165 asm-commons 9.2
1.165.1 Available under license
1.166 fabric8-::-kubernetes-model-::-metrics 5.12.1
1.166.1 Available under license
1.167 fabric8-::-kubernetes-model-::-policy 5.12.1
1.167.1 Available under license
1.168 apache-log4j-api 2.17.2
1.168.1 Available under license
1.169 fabric8-::-kubernetes-model-::-discovery 5.12.1

1.169.1 Available under license
1.170 javabeans-activation-framework-api 1.2.0
1.170.1 Available under license
1.171 snappy-java 1.1.8.4
1.171.1 Available under license
1.172 okhttp 4.10.0
1.172.1 Available under license
1.173 junit 4.13.2
1.173.1 Available under license
1.174 reactive-streams 1.0.4
1.174.1 Available under license
1.175 spring-cloud-sleuth-brave 3.1.3
1.175.1 Available under license
1.176 asm-analysis 9.1
1.176.1 Available under license
1.177 apache-commons-beanutils 1.9.4
1.177.1 Available under license
1.178 io-projectreactor 3.4.21
1.178.1 Available under license
1.179 netty-transport-native-unix-common 4.1.79.Final
1.179.1 Available under license
1.180 objenesis 3.2
1.180.1 Available under license
1.181 asm 9.2
1.181.1 Available under license
1.182 fabric8-::-kubernetes-model-::-coordination 5.12.1
1.182.1 Available under license
1.183 h2 2.1.214
1.183.1 Available under license
1.184 opentelemetry 1.14.0
1.184.1 Available under license
1.185 latencyutils 2.0.3
1.185.1 Available under license
1.186 spring-cloud-context 3.1.3
1.186.1 Available under license
1.187 jboss-logging 3.4.3.Final
1.187.1 Available under license
1.188 spring-boot-validation-starter 2.7.2
1.188.1 Available under license

1.189 spring-security 5.7.2
1.189.1 Available under license
1.190 jackson-bom 2.13.3
1.190.1 Available under license
1.191 opentest4j-opentest4j 1.2.0
1.191.1 Available under license
1.192 launchdarkly-sdk-for-java 5.7.1
1.192.1 Available under license
1.193 checkstyle 9.0.1
1.193.1 Available under license
1.194 jackson-dataformat-yaml 2.13.3
1.194.1 Available under license
1.195 core-functionality-for-the-reactor-netty-library 1.0.21
1.195.1 Available under license
1.196 zstd-jni 1.5.0-4
1.196.1 Available under license
1.197 antlr 4.9.2
1.197.1 Available under license
1.198 okio 3.0.0
1.198.1 Available under license
1.199 elasticsearch-kubernetes-cloud-plugin 5.12.1
1.199.1 Available under license

1.1 java-architecture-for-xml-binding 3.0.0

1.1.1 Available under license :

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Notices for Eclipse Implementation of JAXB

This content is produced and maintained by the Eclipse Implementation of JAXB project.

* Project home: https://projects.eclipse.org/projects/ee4j.jaxb-impl

Trademarks

Eclipse Implementation of JAXB is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0 which is available at http://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaxb-ri
* https://github.com/eclipse-ee4j/jaxb-istack-commons
* https://github.com/eclipse-ee4j/jaxb-dtd-parser
* https://github.com/eclipse-ee4j/jaxb-fi
* https://github.com/eclipse-ee4j/jaxb-stax-ex
* https://github.com/eclipse-ee4j/jax-rpc-ri

Third-party Content

This project leverages the following third party content.

Apache Ant (1.10.2)

* License: Apache-2.0 AND W3C AND LicenseRef-Public-Domain

Apache Ant (1.10.2)

* License: Apache-2.0 AND W3C AND LicenseRef-Public-Domain

Apache Felix (1.2.0)

* License: Apache License, 2.0

args4j (2.33)

* License: MIT License

dom4j (1.6.1)

* License: Custom license based on Apache 1.1

file-management (3.0.0)

* License: Apache-2.0
* Project: https://maven.apache.org/shared/file-management/
* Source:
  https://svn.apache.org/viewvc/maven/shared/tags/file-management-3.0.0/

JUnit (4.12)

* License: Eclipse Public License

JUnit (4.12)

* License: Eclipse Public License

maven-compat (3.5.2)

* License: Apache-2.0
* Project: https://maven.apache.org/ref/3.5.2/maven-compat/
* Source:
  https://mvnrepository.com/artifact/org.apache.maven/maven-compat/3.5.2

maven-core (3.5.2)

* License: Apache-2.0

* Project: https://maven.apache.org/ref/3.5.2/maven-core/index.html
* Source: https://mvnrepository.com/artifact/org.apache.maven/maven-core/3.5.2

maven-plugin-annotations (3.5)

* License: Apache-2.0
* Project: https://maven.apache.org/plugin-tools/maven-plugin-annotations/
* Source:
  https://github.com/apache/maven-plugin-tools/tree/master/maven-plugin-annotations

maven-plugin-api (3.5.2)

* License: Apache-2.0

maven-resolver-api (1.1.1)

* License: Apache-2.0

maven-resolver-api (1.1.1)

* License: Apache-2.0

maven-resolver-connector-basic (1.1.1)

* License: Apache-2.0

maven-resolver-impl (1.1.1)

* License: Apache-2.0

maven-resolver-spi (1.1.1)

* License: Apache-2.0

maven-resolver-transport-file (1.1.1)

* License: Apache-2.0
* Project: https://maven.apache.org/resolver/maven-resolver-transport-file/
* Source:
  https://github.com/apache/maven-resolver/tree/master/maven-resolver-transport-file

maven-resolver-util (1.1.1)

* License: Apache-2.0

maven-settings (3.5.2)

* License: Apache-2.0

* Source:
  https://mvnrepository.com/artifact/org.apache.maven/maven-settings/3.5.2

OSGi Service Platform Core Companion Code (6.0)

* License: Apache License, 2.0

plexus-archiver (3.5)

* License: Apache-2.0
* Project: https://codehaus-plexus.github.io/plexus-archiver/
* Source: https://github.com/codehaus-plexus/plexus-archiver

plexus-io (3.0.0)

* License: Apache-2.0

plexus-utils (3.1.0)

* License: Apache- 2.0 or Apache- 1.1 or BSD or Public Domain or Indiana
  University Extreme! Lab Software License V1.1.1 (Apache 1.1 style)

relaxng-datatype (1.0)

* License: New BSD license

Sax (0.2)

* License: SAX-PD
* Project: http://www.megginson.com/downloads/SAX/
* Source: http://sourceforge.net/project/showfiles.php?group_id=29449

testng (6.14.2)

* License: Apache-2.0 AND (MIT OR GPL-1.0+)
* Project: https://testng.org/doc/index.html
* Source: https://github.com/cbeust/testng

wagon-http-lightweight (3.0.0)

* License: Pending
* Project: https://maven.apache.org/wagon/
* Source:
  https://mvnrepository.com/artifact/org.apache.maven.wagon/wagon-http-lightweight/3.0.0

xz for java (1.8)

* License: LicenseRef-Public-Domain

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.2 opentelemetry-java---io.opentelemetry:opentelemetry-sdk-metrics 1.14.0

1.2.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices

that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not

on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.3 spring-cloud-sleuth-instrumentation 3.1.3

1.3.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2013-2018 the original author or authors.
~
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.
-->

Found in path(s):
* /opt/cola/permits/1519136966_1671710596.3205059/0/spring-cloud-sleuth-instrumentation-3-1-3-jar/META-INF/maven/org.springframework.cloud/spring-cloud-sleuth-instrumentation/pom.xml

1.4 jul-to-slf4j-bridge 1.7.36

1.4.1 Available under license :

No license file was found, but licenses were detected in source scan.

/**
* Copyright (c) 2004-2011 QOS.ch
* All rights reserved.
*
* Permission is hereby granted, free of charge, to any person obtaining
* a copy of this software and associated documentation files (the
* "Software"), to deal in the Software without restriction, including
* without limitation the rights to use, copy, modify, merge, publish,
* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to

* the following conditions:
*
* The above copyright notice and this permission notice shall be
* included in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
* LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
* OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*
*/

Found in path(s):
* /opt/cola/permits/1337069837_1654208675.2587817/0/jul-to-slf4j-1-7-36-sources-jar/org/slf4j/bridge/SLF4JBridgeHandler.java

1.5 brave-instrumentation-spring-rabbitmq 5.13.9

1.5.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity

exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0
* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed

by Square, Inc in the Moshi Library:

* Copyright (C) 2010 Google Inc.
* License: Apache License v2.0
* Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

* License: Apache License v2.0
* Homepage: https://github.com/raphw/weak-lock-free

1.6 testcontainers-jdbc 1.17.1

1.6.1 Available under license :

neo4j:4.4.1-enterprise
ibmcom/db2:11.5.0.0a
The MIT License (MIT)

Copyright (c) 2017 - 2019 G DATA Software AG and other authors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2017 Capital One Services, LLC and other authors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is

furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015-2019 Richard North

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
a
b
mcr.microsoft.com/mssql/server:2017-CU12

1.7 json-library-from-android-sdk 0.0.20131108.vaadin1

1.7.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2010 The Android Open Source Project
*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1089224571_1611250185.24/0/android-json-0-0-20131108-vaadin1-sources-jar/org/json/JSONException.java
* /opt/cola/permits/1089224571_1611250185.24/0/android-json-0-0-20131108-vaadin1-sources-jar/org/json/JSONTokener.java
* /opt/cola/permits/1089224571_1611250185.24/0/android-json-0-0-20131108-vaadin1-sources-jar/org/json/JSONStringer.java
* /opt/cola/permits/1089224571_1611250185.24/0/android-json-0-0-20131108-vaadin1-sources-jar/org/json/JSONObject.java
* /opt/cola/permits/1089224571_1611250185.24/0/android-json-0-0-20131108-vaadin1-sources-jar/org/json/JSONArray.java
* /opt/cola/permits/1089224571_1611250185.24/0/android-json-0-0-20131108-vaadin1-sources-jar/org/json/JSON.java

1.8 snake-yaml 1.30

1.8.1 Available under license :

Found license 'GNU Lesser General Public License' in '// This module is multi-licensed and may be used under the terms // EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal // LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html // GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html // AL, Apache License, V2.0 or later, http://www.apache.org/licenses // BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php * Multi-licensed: EPL / LGPL / GPL / AL / BSD.'
Found license 'Eclipse Public License 1.0' in '// This module is multi-licensed and may be used under the terms // EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal // LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html // GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html // AL, Apache License, V2.0 or later, http://www.apache.org/licenses // BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php * Multi-licensed: EPL / LGPL / GPL / AL / BSD.'

1.9 testcontainers-jdbc-postgresql 1.17.1

1.9.1 Available under license :

neo4j:4.4.1-enterprise
ibmcom/db2:11.5.0.0a
The MIT License (MIT)

Copyright (c) 2017 - 2019 G DATA Software AG and other authors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2017 Capital One Services, LLC and other authors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015-2019 Richard North

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
a
b
mcr.microsoft.com/mssql/server:2017-CU12

1.10 brave-instrumentation-apache-httpasyncclient-v 5.13.9

1.10.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0

* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed
by Square, Inc in the Moshi Library:

* Copyright (C) 2010 Google Inc.
* License: Apache License v2.0
* Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

* License: Apache License v2.0
* Homepage: https://github.com/raphw/weak-lock-free

1.11 zjsonpatch 0.3.0

1.11.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.flipkart.zjsonpatch</groupId>
  <artifactId>zjsonpatch</artifactId>
  <version>0.3.0</version>
  <packaging>jar</packaging>

  <name>zjsonpatch</name>
  <description>Java Library to find / apply JSON Patches according to RFC 6902</description>
  <url>https://github.com/flipkart-incubator/zjsonpatch/</url>

  <scm>
    <connection>scm:git:https://github.com/flipkart-incubator/zjsonpatch.git</connection>
    <developerConnection>scm:git:https://github.com/flipkart-incubator/zjsonpatch.git</developerConnection>
    <tag>HEAD</tag>
    <url>https://github.com/flipkart-incubator/zjsonpatch</url>
  </scm>

  <developers>
    <developer>
      <id>vishwakarma</id>
      <name>Gopi Vishwakarma</name>
      <email>vishwakarma.iiita@gmail.com</email>
    </developer>
  </developers>
```

```
<licenses>
  <license>
    <name>The Apache Software License, Version 2.0</name>
    <url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>
    <distribution>repo</distribution>
  </license>
</licenses>

<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <jackson.version>2.8.5</jackson.version>
</properties>

<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>2.3.1</version>
      <configuration>
        <source>1.6</source>
        <target>1.6</target>
        <showDeprecation>true</showDeprecation>
        <showWarnings>true</showWarnings>
      </configuration>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-source-plugin</artifactId>
      <version>2.2.1</version>
      <executions>
        <execution>
          <id>attach-sources</id>
          <goals>
            <goal>jar-no-fork</goal>
          </goals>
        </execution>
      </executions>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-javadoc-plugin</artifactId>
      <version>2.9.1</version>
      <executions>
        <execution>
          <id>attach-javadocs</id>
          <goals>
```

```
                <goal>jar</goal>
              </goals>
            </execution>
          </executions>
        </plugin>
        <plugin>
          <groupId>org.sonatype.plugins</groupId>
          <artifactId>nexus-staging-maven-plugin</artifactId>
          <version>1.6.3</version>
          <extensions>true</extensions>
          <configuration>
            <serverId>ossrh</serverId>
            <nexusUrl>https://oss.sonatype.org/</nexusUrl>
            <autoReleaseAfterClose>true</autoReleaseAfterClose>
          </configuration>
        </plugin>
        <plugin>
          <groupId>org.apache.maven.plugins</groupId>
          <artifactId>maven-gpg-plugin</artifactId>
          <version>1.4</version>
          <executions>
            <execution>
              <id>sign-artifacts</id>
              <phase>verify</phase>
              <goals>
                <goal>sign</goal>
              </goals>
            </execution>
          </executions>
        </plugin>
        <plugin>
          <groupId>org.apache.maven.plugins</groupId>
          <artifactId>maven-release-plugin</artifactId>
          <version>2.5</version>
          <configuration>
            <autoVersionSubmodules>true</autoVersionSubmodules>
            <useReleaseProfile>false</useReleaseProfile>
            <releaseProfiles>release</releaseProfiles>
            <goals>deploy</goals>
          </configuration>
        </plugin>
      </plugins>
    </build>
    <dependencies>
      <dependency>
        <groupId>com.fasterxml.jackson.core</groupId>
        <artifactId>jackson-databind</artifactId>
        <version>${jackson.version}</version>
```

```
    </dependency>
    <dependency>
      <groupId>com.fasterxml.jackson.core</groupId>
      <artifactId>jackson-core</artifactId>
      <version>${jackson.version}</version>
    </dependency>
    <dependency>
      <groupId>com.google.guava</groupId>
      <artifactId>guava</artifactId>
      <version>18.0</version>
    </dependency>
    <dependency>
      <scope>test</scope>
      <groupId>commons-io</groupId>
      <artifactId>commons-io</artifactId>
      <version>2.4</version>
    </dependency>
    <dependency>
      <scope>test</scope>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.12</version>
    </dependency>
    <dependency>
      <groupId>org.apache.commons</groupId>
      <artifactId>commons-collections4</artifactId>
      <version>4.1   </version>
    </dependency>
  </dependencies>
  <distributionManagement>
    <snapshotRepository>
      <id>ossrh</id>
      <url>https://oss.sonatype.org/content/repositories/snapshots</url>
    </snapshotRepository>
    <repository>
      <id>ossrh</id>
      <url>https://oss.sonatype.org/service/local/staging/deploy/maven2/</url>
    </repository>
  </distributionManagement>
</project>
```

Found in path(s):
* /opt/cola/permits/1288519755_1647351836.02/0/zjsonpatch-0-3-0-jar/META-INF/maven/com.flipkart.zjsonpatch/zjsonpatch/pom.xml

1.12 jackson-core 2.13.3

1.12.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0
To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,

including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf

of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.13 fabric8-::-kubernetes-model-::-core 5.12.1

1.13.1 Available under license :

No license file was found, but licenses were detected in source scan.

/**
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/WatchEvent.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/GenericKubernetesResource.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/KubernetesList.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/Pluralize.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/Quantity.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/KubernetesResourceMappingProvider.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/HasMetadataComparator.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/IntOrString.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/internal/KubernetesDeserializer.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/KubernetesResource.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/DefaultKubernetesResourceList.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/KubernetesResourceList.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/internal/InternalResourceMappingProvider.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/MicroTimeSerDes.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/GenericKubernetesResourceList.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/Namespaced.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/DeletionPropagation.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/HasMetadata.java
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/Duration.java

No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

       http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-core/pom.xml
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340815874_1654860883.3965907/0/kubernetes-model-core-5-12-1-sources-jar/manifest.vm

1.14 netty-project 4.1.79.Final

1.14.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Implementation-Title: Netty/All-in-One
Bundle-Description: Netty is an asynchronous event-driven network appl
ication framework for rapid development of maintainable high perfo
rmance protocol servers and clients.
Automatic-Module-Name: io.netty.all
Bundle-License: https://www.apache.org/licenses/LICENSE-2.0
Bundle-SymbolicName: io.netty.all
Implementation-Version: 4.1.79.Final
Built-By: chris
Bnd-LastModified: 1657574625518
Bundle-ManifestVersion: 2
Implementation-Vendor-Id: io.netty
Bundle-DocURL: https://netty.io/
Bundle-Vendor: The Netty Project
Import-Package: sun.nio.ch;resolution:=optional,org.eclipse.jetty.npn;
version="[1,2)";resolution:=optional,org.eclipse.jetty.alpn;version="
[1,2)";resolution:=optional
Tool: Bnd-2.4.1.201501161923
Implementation-Vendor: The Netty Project
Bundle-Name: Netty/All-in-One
Bundle-Version: 4.1.79.Final
Created-By: Apache Maven Bundle Plugin
Build-Jdk: 1.8.0_312
Implementation-URL: https://netty.io/netty-all/

Found in path(s):
* /opt/cola/permits/1519137082_1671725178.437221/0/netty-all-4-1-79-final-jar/META-INF/MANIFEST.MF
No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->

Found in path(s):
* /opt/cola/permits/1519137082_1671725178.437221/0/netty-all-4-1-79-final-jar/META-INF/maven/io.netty/netty-all/pom.xml

1.15 jackson-datatype-jsr310 2.13.3

1.15.1 Available under license :

This copy of Jackson JSON processor streaming parser/generator is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.16 xmlunit-xmlunit-core 2.9.0

1.16.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--
This file is licensed to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->

Found in path(s):
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/META-INF/maven/org.xmlunit/xmlunit-core/pom.xml
No license file was found, but licenses were detected in source scan.

/*
This file is licensed to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/TransformationBuilderBase.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/validation/Validator.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/DifferenceEvaluator.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/xpath/JAXPXPathEngine.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/DifferenceEvaluators.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/NodeFilters.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ComparisonListenerSupport.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/package-info.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/util/TransformerFactoryConfigurer.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/input/WhitespaceStrippedSource.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/util/Nodes.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/util/Predicate.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/xpath/XPathEngine.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ComparisonListener.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ComparisonController.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/DOMDifferenceEngine.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/DefaultNodeMatcher.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/validation/ParsingValidator.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/DefaultComparisonFormatter.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-

jar/org/xmlunit/util/Mapper.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/xpath/package-info.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/validation/package-info.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/input/ElementContentWhitespaceStrippedSource.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/MultiLevelByNameAndTextSelector.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/RecursiveXPathBuilder.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/input/package-info.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/util/IsNullPredicate.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/input/CommentLessSource.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/javax_jaxb/package-info.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/validation/ValidationResult.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/transform/Transformation.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ComparisonResult.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ComparisonControllers.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/AbstractDifferenceEngine.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/XMLUnitException.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ElementSelector.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ElementSelectors.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/Comparison.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ComparisonType.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/validation/ValidationHandler.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/AbstractTransformationBuilder.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/input/NormalizedSource.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/Diff.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/util/DocumentBuilderFactoryConfigurer.java

* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ByNameAndTextRecSelector.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ChildNodeXPathContextProvider.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/DefaultConditionalSelectorBuilder.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/util/Linqy.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/Transform.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/validation/Languages.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/transform/package-info.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/package-info.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/NodeMatcher.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/util/IterableNodeList.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/util/package-info.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/ConfigurationException.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/package-info.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/validation/JAXPValidator.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/DifferenceEngine.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/input/WhitespaceNormalizedSource.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/XPathContext.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/util/Convert.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/Input.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/validation/ValidationProblem.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/Difference.java
No license file was found, but licenses were detected in source scan.

/*
 This file is licensed to You under the Apache License, Version 2.0
 (the "License"); you may not use this file except in compliance with
 the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/JaxbBuilder.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/diff/ComparisonFormatter.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/DifferenceEngineConfigurer.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/JaxbBuilderFactory.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/javax_jaxb/JavaxJaxbBuilder.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/DiffBuilder.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/javax_jaxb/DefaultJaxbBuilderFactory.java
* /opt/cola/permits/1344549174_1655241028.106165/0/xmlunit-core-2-9-0-sources-jar/org/xmlunit/builder/JaxbBuilderFactoryLocator.java

1.17 fabric8-::-kubernetes-model-::-node 5.12.1

1.17.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

Copyright (C) 2015 Red Hat, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and
limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815880_1654860784.2252932/0/kubernetes-model-node-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-node/pom.xml
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340815880_1654860784.2252932/0/kubernetes-model-node-5-12-1-sources-jar/manifest.vm

1.18 apache-log4j-jul-adapter 2.17.2

1.18.1 Available under license :

Apache Log4j JUL Adapter
Copyright 1999-2022 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and

limitations under the License.

1.19 junit-platform-junit-platform-commons 1.8.2

1.19.1 Available under license :

Eclipse Public License - v 2.0
==========

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:
* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **b)** in the case of each subsequent Contributor:
 * **i)** changes to the Program, and
 * **ii)** additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer

of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

3.1 If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
 * **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied,

including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 * **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 * **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
 * **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
* **b)** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION,

ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are

created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
Open Source Licenses
==========

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for
these subcomponents is subject to the terms and conditions of the
subcomponent's license, as noted in the LICENSE-<subcomponent>.md
files.

1.20 fabric8-::-kubernetes-model-::-extensions 5.12.1

1.20.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815636_1654860835.7045515/0/kubernetes-model-extensions-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-extensions/pom.xml
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340815636_1654860835.7045515/0/kubernetes-model-extensions-5-12-1-sources-jar/manifest.vm

1.21 netty/transport/classes/epoll 4.1.79.Final

1.21.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~   https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->

Found in path(s):
* /opt/cola/permits/1519136991_1671720838.5587008/0/netty-transport-native-epoll-4-1-79-final-sources-1-jar/META-INF/maven/io.netty/netty-transport-native-epoll/pom.xml
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519136991_1671720838.5587008/0/netty-transport-native-epoll-4-1-79-final-sources-1-jar/netty_epoll_native.c
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519136991_1671720838.5587008/0/netty-transport-native-epoll-4-1-79-final-sources-1-jar/netty_epoll_linuxsocket.h
* /opt/cola/permits/1519136991_1671720838.5587008/0/netty-transport-native-epoll-4-1-79-final-sources-1-jar/netty_epoll_linuxsocket.c

1.22 java-architecture-for-xml-binding 2.3.1

1.22.1 Available under license :

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor

Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their

nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty

protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.23 spring-boot-actuator 2.7.2

1.23.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation

source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and

may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,

defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Spring Boot 2.7.2
Copyright (c) 2012-2022 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

1.24 fabric8-::-kubernetes-model-::-common 5.12.1

1.24.1 Available under license :

No license file was found, but licenses were detected in source scan.

/**
* Copyright (C) 2015 Red Hat, Inc.
*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/jackson/JsonUnwrappedDeserializer.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/Scope.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/util/Helper.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/Version.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/Generated.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/ShortNames.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/jackson/UnwrappedTypeResolverBuilder.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/Singular.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/LabelSelector.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/Group.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/Kind.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/SpecReplicas.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/StatusReplicas.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/Plural.java
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/io/fabric8/kubernetes/model/annotation/PrinterColumn.java
No license file was found, but licenses were detected in source scan.

<!--

Copyright (C) 2015 Red Hat, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815941_1654860906.0784943/0/kubernetes-model-common-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-common/pom.xml

1.25 fabric8-::-kubernetes-model-::-admission-registration,-authentication-and-authorization 5.12.1

1.25.1 Available under license :

No license file was found, but licenses were detected in source scan.

/**
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340815643_1654860986.7683268/0/kubernetes-model-admissionregistration-5-12-1-sources-

jar/io/fabric8/kubernetes/api/model/admission/v1/AdmissionRequest.java
* /opt/cola/permits/1340815643_1654860986.7683268/0/kubernetes-model-admissionregistration-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/admission/v1beta1/AdmissionRequest.java
No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

       http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815643_1654860986.7683268/0/kubernetes-model-admissionregistration-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-admissionregistration/pom.xml
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340815643_1654860986.7683268/0/kubernetes-model-admissionregistration-5-12-1-sources-jar/manifest.vm

1.26 spring-webflux 5.3.22

1.26.1 Available under license :

Spring Framework 5.3.22
Copyright (c) 2002-2022 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for
these subcomponents is subject to the terms and conditions of the
subcomponent's license, as noted in the license.txt file.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

==========

SPRING FRAMEWORK 5.3.22 SUBCOMPONENTS:

Spring Framework 5.3.22 includes a number of subcomponents
with separate copyright notices and license terms. The product that
includes this file does not necessarily use all the open source
subcomponents referred to below. Your use of the source
code for these subcomponents is subject to the terms and
conditions of the following licenses.

>>> ASM 9.1 (org.ow2.asm:asm:9.1, org.ow2.asm:asm-commons:9.1):

Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright

notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1999-2009, OW2 Consortium <https://www.ow2.org/>

>>> CGLIB 3.3 (cglib:cglib:3.3):

Per the LICENSE file in the CGLIB JAR distribution downloaded from
https://github.com/cglib/cglib/releases/download/RELEASE_3_3_0/cglib-3.3.0.jar,
CGLIB 3.3 is licensed under the Apache License, version 2.0, the text of which
is included above.

>>> Objenesis 3.2 (org.objenesis:objenesis:3.2):

Per the LICENSE file in the Objenesis ZIP distribution downloaded from
http://objenesis.org/download.html, Objenesis 3.2 is licensed under the
Apache License, version 2.0, the text of which is included above.

Per the NOTICE file in the Objenesis ZIP distribution downloaded from
http://objenesis.org/download.html and corresponding to section 4d of the
Apache License, Version 2.0, in this case for Objenesis:

Objenesis
Copyright 2006-2019 Joe Walnes, Henri Tremblay, Leonardo Mesquita

==========

To the extent any open source components are licensed under the EPL and/or other similar licenses that require the source code and/or modifications to source code to be made available (as would be noted above), you may obtain a copy of the source code corresponding to the binaries for such open source components and modifications thereto, if any, (the "Source Files"), by downloading the Source Files from https://spring.io/projects, Pivotal's website at https://network.pivotal.io/open-source, or by sending a request, with your name and address to: Pivotal Software, Inc., 875 Howard Street, 5th floor, San Francisco, CA 94103, Attention: General Counsel. All such requests should clearly specify: OPEN SOURCE FILES REQUEST, Attention General Counsel. Pivotal can mail a copy of the Source Files to you on a CD or equivalent physical medium.

This offer to obtain a copy of the Source Files is valid for three years from the date you acquired this Software product. Alternatively, the Source Files may accompany the Software.

1.27 spring-cloud-starter-sleuth 3.1.3

1.27.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2013-2018 the original author or authors.
~
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~
~   https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.
-->

Found in path(s):
* /opt/cola/permits/1519137062_1671710577.1414895/0/spring-cloud-starter-sleuth-3-1-3-jar/META-INF/maven/org.springframework.cloud/spring-cloud-starter-sleuth/pom.xml

1.28 asm-tree 9.1

1.28.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bundle-DocURL: http://asm.ow2.org
Bundle-License: BSD-3-Clause;link=https://asm.ow2.io/LICENSE.txt
Bundle-ManifestVersion: 2
Bundle-Name: org.objectweb.asm.tree
Bundle-RequiredExecutionEnvironment: J2SE-1.5
Bundle-SymbolicName: org.objectweb.asm.tree
Bundle-Version: 9.1.0
Export-Package: org.objectweb.asm.tree;version="9.1";uses:="org.object
web.asm"
Implementation-Title: Tree API of ASM, a very small and fast Java byte
code manipulation framework
Implementation-Version: 9.1
Import-Package: org.objectweb.asm;version="[9.1,10)"
Module-Requires: org.objectweb.asm;transitive=true

Found in path(s):
* /opt/cola/permits/1183890441_1627493647.83/0/asm-tree-9-1-jar/META-INF/MANIFEST.MF

1.29 postgresql-jdbc-driver-pgjdbc 42.3.3

1.29.1 Available under license :

Copyright 2017, OnGres.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2019, OnGres.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 1997, PostgreSQL Global Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

Additional License files can be found in the 'licenses' folder located in the same directory as the LICENSE file (i.e. this file)

- Software produced outside the ASF which is available under other licenses (not Apache-2.0)

BSD-2-Clause
* com.ongres.scram:client:2.1
* com.ongres.scram:common:2.1
* com.ongres.stringprep:saslprep:1.1
* com.ongres.stringprep:stringprep:1.1

1.30 spring-boot 2.7.2

1.30.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Spring Boot 2.7.2
Copyright (c) 2012-2022 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

1.31 javassist 3_26_0_ga

1.31.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that

any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a

"work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy,

and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library

with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the
source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not
compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the
Library, but is designed to work with the Library by being compiled or
linked with it, is called a "work that uses the Library". Such a
work, in isolation, is not a derivative work of the Library, and
therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library
creates an executable that is a derivative of the Library (because it
contains portions of the Library), rather than a "work that uses the
library". The executable is therefore covered by this License.
Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file
that is part of the Library, the object code for the work may be a
derivative work of the Library even though the source code is not.

Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if

the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any

attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing

to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

/*
* Javassist, a Java-bytecode translator toolkit.
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.
*
* The contents of this file are subject to the Mozilla Public License Version
* 1.1 (the "License"); you may not use this file except in compliance with
* the License. Alternatively, the contents of this file may be used under
* the terms of the GNU Lesser General Public License Version 2.1 or later,
* or the Apache License Version 2.0.
*
* Software distributed under the License is distributed on an "AS IS" basis,
* WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
* for the specific language governing rights and limitations under the
* License.
*/

1.32 fabric8-::-kubernetes-model-::-certificates 5.12.1

1.32.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

Copyright (C) 2015 Red Hat, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340816165_1654860920.0299604/0/kubernetes-model-certificates-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-certificates/pom.xml
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340816165_1654860920.0299604/0/kubernetes-model-certificates-5-12-1-sources-jar/manifest.vm

1.33 error_prone_annotations 2.7.1

1.33.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Error Prone Authors.

*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/NoAllocation.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/LockMethod.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/UnlockMethod.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The Error Prone Authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/OverridingMethodsMustInvokeSuper.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/DoNotCall.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/GuardedBy.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/CheckReturnValue.java

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2016 The Error Prone Authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/FormatString.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/CompatibleWith.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/RestrictedApi.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/MustBeClosed.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/DoNotMock.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/FormatMethod.java

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2015 The Error Prone Authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/Var.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/LazyInit.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/IncompatibleModifiers.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/RequiredModifiers.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/Immutable.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/CompileTimeConstant.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/SuppressPackageLocation.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/ForOverride.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/CanIgnoreReturnValue.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2021 The Error Prone Authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/InlineMeValidationDisabled.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/InlineMe.java

1.34 aspectj-runtime 1.9.7

1.34.1 Available under license :

No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 1999-2001 Xerox Corporation,
*   2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Xerox/PARC   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/CFlowStack.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/MethodSignatureImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/StringMaker.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/Signature.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/NoAspectBoundException.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/ConstructorSignature.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/FieldSignature.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/ConstructorSignatureImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/SourceLocationImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/CFlowPlusState.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/CodeSignatureImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/CatchClauseSignature.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/InitializerSignatureImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/CFlow.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/AdviceSignature.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/JoinPoint.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/SignatureImpl.java

* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/CodeSignature.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/MethodSignature.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/JoinPointImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/Conversions.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/MemberSignature.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/SourceLocation.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/FieldSignatureImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/PerObjectMap.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/AdviceSignatureImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/CatchClauseSignatureImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/MemberSignatureImpl.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Wes Isberg   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/internal/lang/PlaceHolder.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2004 IBM Corporation
*
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at

* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadCounter.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/CFlowCounter.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadStackFactoryImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadStack.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadStackFactoryImpl11.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadStackFactory.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 1999-2001 Xerox Corporation,
*   2002 Palo Alto Research Center, Incorporated (PARC),
*   2004 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Xerox/PARC   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/SoftException.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*

* Contributors:
* initial implementation Alexandre Vasseur
*****/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/Aspect.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/Before.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/Pointcut.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/AfterReturning.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclarePrecedence.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/After.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/Around.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/AfterThrowing.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/ProceedingJoinPoint.java
No license file was found, but licenses were detected in source scan.

/**
* Copyright (c) 2009 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* initial implementation Andy Clement
**/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclareMixin.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at

* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Wes Isberg   initial implementation
*   Andy Clement   fleshed out to match SuppressWarnings
* **/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/SuppressAjWarnings.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors.
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Adrian Colyer   initial implementation
***/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/AdviceName.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2004 IBM Corporation
*
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement   initial implementation
*   Copied from bits of original CFlowStack
* **/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadStackImpl11.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadCounterImpl11.java

No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* initial implementation Adrian Colyer
***/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclareParents.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Adrian Colyer Initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/InterTypeFieldDeclaration.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/InterTypeConstructorDeclarationImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/DeclareSoftImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/DeclareAnnotation.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/PerClause.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/DeclarePrecedence.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/SignaturePattern.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/DeclareSoft.java

* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/AjTypeSystem.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/PointcutImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/DeclareErrorOrWarning.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcDeclareSoft.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/Advice.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/TypePattern.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/PointcutExpression.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/TypePatternImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcITD.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/AjType.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcDeclareEoW.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcDeclarePrecedence.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclareWarning.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/InterTypeConstructorDeclaration.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcDeclareAnnotation.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/DeclareErrorOrWarningImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/TypePatternBasedPerClause.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/PerClauseKind.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/SignaturePatternImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/DeclareParents.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/AdviceImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/DeclareParentsImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/InterTypeDeclaration.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/PointcutExpressionImpl.java

* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/DeclarePrecedenceImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/InterTypeMethodDeclarationImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/InterTypeMethodDeclaration.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/NoSuchAdviceException.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/StringToType.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/NoSuchPointcutException.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/Pointcut.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/PointcutBasedPerClause.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcDeclareParents.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/InterTypeFieldDeclarationImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/InterTypeDeclarationImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/DeclareAnnotationImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/TypePatternBasedPerClauseImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/AdviceKind.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcPrivileged.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/PointcutBasedPerClauseImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/AjTypeImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/PerClauseImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclareError.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 1999-2001 Xerox Corporation,
*   2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt

*
* Contributors:
*   Xerox/PARC   initial implementation
*   Alex Vasseur   wired up for @AJ proceeding
*   Andy Clement 23-06-06 added extras for @AJ
* **/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/internal/AroundClosure.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2006 IBM Corporation and others.
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement - initial implementation
***/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/LockSignatureImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/LockSignature.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/UnlockSignatureImpl.java
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/UnlockSignature.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2014 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/RequiredTypes.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* initial implementation              Alexandre Vasseur
* generic signature update            Adrian Colyer
*******************************************************************************/
```

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/Aspects.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 1999-2001 Xerox Corporation,
*               2002-2018 Palo Alto Research Center, Incorporated (PARC), Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Xerox/PARC    initial implementation
*    Alex Vasseur    new factory methods for variants of JP
*  Abraham Nevado    new factory methods for collapsed SJPs
*    Andy Clement    new factory methods that rely on LDC <class>
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/runtime/reflect/Factory.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
* Copyright (c) 2006 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* variant of Aspects in the aspectj5rt project - this one isn't Java5   - Andy Clement
```

**********/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/Aspects14.java
No license file was found, but licenses were detected in source scan.

/**********
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Alexandre Vasseur   initial implementation
**********/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclareAnnotation.java
No license file was found, but licenses were detected in source scan.

/**********
* Copyright (c) 2013 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* initial implementation   Andy Clement
**********/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/annotation/control/CodeGenerationHint.java
No license file was found, but licenses were detected in source scan.

/* **********
* Copyright (c) 1999-2001 Xerox Corporation,
*   2002 Palo Alto Research Center, Incorporated (PARC),
*   2006 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at

* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Xerox/PARC   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1183891981_1654861569.6063452/0/aspectjrt-1-9-7-sources-jar/org/aspectj/lang/reflect/InitializerSignature.java

1.35 spring-boot-reactor-netty-starter 2.7.2

1.35.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Spring Boot 2.7.2
Copyright (c) 2012-2022 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

1.36 mac-os 4.1.80.Final

1.36.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136954_1671710585.2904873/0/netty-resolver-dns-classes-macos-4-1-80-final-sources-2-jar/io/netty/resolver/dns/macos/DnsResolver.java
* /opt/cola/permits/1519136954_1671710585.2904873/0/netty-resolver-dns-classes-macos-4-1-80-final-sources-2-jar/io/netty/resolver/dns/macos/package-info.java
* /opt/cola/permits/1519136954_1671710585.2904873/0/netty-resolver-dns-classes-macos-4-1-80-final-sources-2-jar/io/netty/resolver/dns/macos/MacOSDnsServerAddressStreamProvider.java
No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2021 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~   https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->

Found in path(s):
* /opt/cola/permits/1519136954_1671710585.2904873/0/netty-resolver-dns-classes-macos-4-1-80-final-sources-2-jar/META-INF/maven/io.netty/netty-resolver-dns-classes-macos/pom.xml

1.37 testcontainers-core 1.17.1

1.37.1 Available under license :

neo4j:4.4.1-enterprise
ibmcom/db2:11.5.0.0a
The MIT License (MIT)

Copyright (c) 2017 - 2019 G DATA Software AG and other authors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2017 Capital One Services, LLC and other authors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015-2019 Richard North

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

a

b

mcr.microsoft.com/mssql/server:2017-CU12

1.38 zipkin-v2 2.23.2

1.38.1 Available under license :

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This product contains a modified part of Gson, distributed by Google:

* License: Apache License v2.0
* Homepage: https://github.com/google/gson

This product contains a modified part of Guava, distributed by Google:

* License: Apache License v2.0
* Homepage: https://github.com/google/guava

This product contains a modified part of Okio, distributed by Square:

* License: Apache License v2.0
* Homepage: https://github.com/square/okio

1.39 json-path 2.7.0

1.39.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013 the original author or authors.

*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/internal/path/json/mapping/JsonPathJackson2ObjectDeserializer.groovy
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/path/json/config/JsonPathConfig.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/mapper/factory/Jackson1ObjectMapperFactory.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/mapper/factory/DefaultJackson2ObjectMapperFactory.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/internal/path/json/mapping/JsonObjectDeserializer.groovy
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/mapper/factory/DefaultJackson1ObjectMapperFactory.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/path/json/config/JsonParserType.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/internal/path/json/JSONAssertion.groovy
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/internal/path/json/ConfigurableJsonSlurper.groovy
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/mapper/factory/DefaultGsonObjectMapperFactory.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/mapper/factory/Jackson2ObjectMapperFactory.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/path/json/JsonPath.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/path/json/exception/JsonPathException.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/internal/path/json/JsonPrettifier.groovy
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/internal/path/json/mapping/JsonPathJackson1ObjectDeserializer.groovy
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/mapper/factory/GsonObjectMapperFactory.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-

jar/com/jayway/restassured/path/json/mapping/JsonPathObjectDeserializer.java
* /opt/cola/permits/1343421013_1655119268.5908961/0/json-path-2-7-0-sources-1-jar/com/jayway/restassured/internal/path/json/mapping/JsonPathGsonObjectDeserializer.groovy

1.40 hibernate-validator 6.2.3.Final

1.40.1 Available under license :

Apache-2.0

1.41 apache-commons-collections 3.2.2

1.41.1 Available under license :

Apache Commons Collections
Copyright 2001-2015 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.42 okhttp 4.9.3

1.42.1 Available under license :

Note that publicsuffixes.gz is compiled from The Public Suffix List:
https://publicsuffix.org/list/public_suffix_list.dat

It is subject to the terms of the Mozilla Public License, v. 2.0:
https://mozilla.org/MPL/2.0/

1.43 checker-qual 3.12.0

1.43.1 Available under license :

Checker Framework qualifiers
Copyright 2004-present by the Checker Framework developers

MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.44 fabric8-::-kubernetes-model-::-flowcontrol 5.12.1

1.44.1 Available under license :

No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340816228_1654861754.4785826/0/kubernetes-model-flowcontrol-5-12-1-sources-jar/manifest.vm
No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340816228_1654861754.4785826/0/kubernetes-model-flowcontrol-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-flowcontrol/pom.xml

1.45 fabric8-::-kubernetes-model-::-networking 5.12.1

1.45.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815668_1654860806.5167248/0/kubernetes-model-networking-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-networking/pom.xml
No license file was found, but licenses were detected in source scan.

```
*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*         http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
```

Found in path(s):
* /opt/cola/permits/1340815668_1654860806.5167248/0/kubernetes-model-networking-5-12-1-sources-jar/manifest.vm

1.46 docker-java-transport 3.2.13

1.46.1 Available under license :

Apache-2.0

1.47 byte-buddy-agent 1.12.12

1.47.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Copyright ${project.inceptionYear} - ${current.year} ${copyright.holder}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.48 asm 9.1

1.48.1 Available under license :

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/signature/package.html
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/package.html
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/TypePath.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Frame.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/RecordComponentWriter.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/AnnotationVisitor.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ModuleVisitor.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ModuleWriter.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/CurrentFrame.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/AnnotationWriter.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ConstantDynamic.java

* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ClassWriter.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/signature/SignatureVisitor.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Attribute.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Opcodes.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Handle.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/MethodTooLargeException.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/signature/SignatureReader.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/TypeReference.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Context.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Label.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Type.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/signature/SignatureWriter.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/FieldVisitor.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Handler.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/SymbolTable.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ClassReader.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Edge.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/RecordComponentVisitor.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Constants.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Symbol.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/FieldWriter.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/MethodWriter.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/MethodVisitor.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ClassTooLargeException.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ByteVector.java
* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ClassVisitor.java

1.49 j2objc-annotations 1.3

1.49.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/RetainedWith.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/Property.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/LoopTranslation.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/ObjectiveCName.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/ReflectionSupport.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/RetainedLocalRef.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/J2ObjCIncompatible.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 Google Inc. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/Weak.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/AutoreleasePool.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/WeakOuter.java

1.50 spring-boot-log4j2-starter 2.7.2

1.50.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Spring Boot 2.7.2
Copyright (c) 2012-2022 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

1.51 jackson-annotations 2.13.3

1.51.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.52 spring-boot-test-auto-configure 2.7.2

1.52.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to

communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of

the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

Spring Boot 2.7.2

Copyright (c) 2012-2022 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0 (the "License"). You may not use this product except in compliance with the License.

1.53 brave-instrumentation-kafka-streams 5.13.9

1.53.1 Available under license :

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0
* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed
by Square, Inc in the Moshi Library:

* Copyright (C) 2010 Google Inc.
* License: Apache License v2.0
* Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

* License: Apache License v2.0

* Homepage: https://github.com/raphw/weak-lock-free

1.54 jooq-meta-extensions 3.16.4

1.54.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

  <modelVersion>4.0.0</modelVersion>

  <parent>
    <groupId>org.jooq</groupId>
    <artifactId>jooq-parent</artifactId>
    <version>3.16.4</version>
  </parent>

  <artifactId>jooq-meta-extensions</artifactId>
  <name>jOOQ Meta Extensions</name>

  <licenses>
    <license>
      <name>Apache License, Version 2.0</name>
      <url>http://www.jooq.org/inc/LICENSE.txt</url>
      <distribution>repo</distribution>
    </license>









  </licenses>

  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-jar-plugin</artifactId>
        <configuration>
          <archive>
            <manifestEntries>
              <Automatic-Module-Name>org.jooq.meta.extensions</Automatic-Module-Name>
            </manifestEntries>
          </archive>
```

```
            </configuration>
          </plugin>
        </plugins>
    </build>

    <dependencies>
        <dependency>
            <groupId>org.jooq</groupId>
            <artifactId>jooq-meta</artifactId>
        </dependency>
        <dependency>
            <groupId>com.h2database</groupId>
            <artifactId>h2</artifactId>
        </dependency>









    </dependencies>
</project>
```

Found in path(s):
* /opt/cola/permits/1340815815_1654719674.1983817/0/jooq-meta-extensions-3-16-4-sources-jar/META-INF/maven/org.jooq/jooq-meta-extensions/pom.xml
No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*  http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Other licenses:
* -----------------------------------------------------------------------------
* Commercial licenses for this work are available. These replace the above
* ASL 2.0 and offer limited warranties, support, maintenance, and commercial
```

* database integrations.
*
* For more information, please visit: http://www.jooq.org/licenses
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*/

Found in path(s):
* /opt/cola/permits/1340815815_1654719674.1983817/0/jooq-meta-extensions-3-16-4-sources-jar/org/jooq/meta/extensions/AbstractInterpretingDatabase.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Other licenses:
* ---
* Commercial licenses for this work are available. These replace the above
* ASL 2.0 and offer limited warranties, support, maintenance, and commercial
* database integrations.
*
* For more information, please visit: http://www.jooq.org/licenses
*
*

*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*/
/**
* The DDL database.
* <p>
* This meta data source parses a set of SQL scripts, translates them to the H2
* dialect and runs them on an in-memory H2 database before reverse engineering
* the outcome.
* <p>
* The SQL scripts are located in the <code>scripts</code> scripts property
* available from {@link #getProperties()}.
*
* @author Lukas Eder
*/

Found in path(s):
* /opt/cola/permits/1340815815_1654719674.1983817/0/jooq-meta-extensions-3-16-4-sources-jar/org/jooq/meta/extensions/ddl/DDLDatabase.java

1.55 junit-jupiter-junit-jupiter-api 5.8.2

1.55.1 Available under license :

Eclipse Public License - v 2.0
==========

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:
* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **b)** in the case of each subsequent Contributor:

 * **i)** changes to the Program, and
 * **ii)** additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

3.1 If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
 * **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 * **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 * **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
 * **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
* **b)** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or

enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
Open Source Licenses
==========

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for

these subcomponents is subject to the terms and conditions of the
subcomponent's license, as noted in the LICENSE-<subcomponent>.md
files.

1.56 spring-retry 1.3.1

1.56.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/support/RetryTemplate.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/ExponentialBackOffPolicy.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006-2014 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/UniformRandomBackOffPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/FixedBackOffPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/context/RetryContextSupport.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/interceptor/Retryable.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/interceptor/RetryInterceptorBuilder.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/util/SimpleMethodInvoker.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/ThreadWaitSleeper.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/annotation/RetryConfiguration.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/BinaryExceptionClassifierBuilder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015-2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/ExpressionRetryPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/support/RetryTemplateBuilder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2014 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and

* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/util/MethodInvokerUtils.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014-2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/annotation/Retryable.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006-2012 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/ExponentialRandomBackOffPolicy.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006-2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/interceptor/StatefulRetryOperationsInterceptor.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/interceptor/MethodInvocationRetryCallback.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/BinaryExceptionClassifier.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/MaxAttemptsRetryPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/BinaryExceptionClassifierRetryPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/listener/MethodInvocationRetryListenerSupport.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/SimpleRetryPolicy.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006-2013 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and

* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/SubclassClassifier.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2008 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/util/MethodInvoker.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/util/AnnotationMethodResolver.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/util/MethodResolver.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2018 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/annotation/CircuitBreaker.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2012-2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/stats/DefaultRetryStatistics.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/stats/RetryStatisticsFactory.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/stats/ExponentialAverageRetryStatistics.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/stats/DefaultRetryStatisticsFactory.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/stats/MutableRetryStatistics.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/stats/StatisticsRepository.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/stats/DefaultStatisticsRepository.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/stats/StatisticsListener.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2014-2018 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
```

*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/annotation/AnnotationAwareRetryOperationsInterceptor.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/CircuitBreakerRetryPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/interceptor/FixedKeyGenerator.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012-2013 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/annotation/Recover.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/annotation/Backoff.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/annotation/EnableRetry.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013-2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/annotation/RecoverAnnotationRecoveryHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006-2007 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/interceptor/RetryOperationsInterceptor.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006-2007 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/BackOffPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/support/RetrySynchronizationManager.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/RetryCallback.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/interceptor/MethodArgumentsKeyGenerator.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/Sleeper.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/RetryCacheCapacityExceededException.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/RetryException.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/RetryContextCache.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/ObjectWaitSleeper.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/NoBackOffPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/support/RetrySimulation.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/BackOffInterruptedException.java

* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/PatternMatchingClassifier.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/RetryPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/ExhaustedRetryException.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/support/DefaultRetryState.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/NeverRetryPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/RetryListener.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/RetryState.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/BackToBackPatternClassifier.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/Classifier.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/PatternMatcher.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/StatelessBackOffPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/RetryOperations.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/BackOffContext.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/backoff/SleepingBackOffPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/ExceptionClassifierRetryPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/CompositeRetryPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/support/RetrySimulator.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/ClassifierSupport.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/MapRetryContextCache.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/listener/RetryListenerSupport.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/SoftReferenceMapRetryContextCache.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/interceptor/NewMethodArgumentsIdentifier.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/AlwaysRetryPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/annotation/Classifier.java

* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/classify/ClassifierAdapter.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/RetryContext.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/TerminatedRetryException.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/policy/TimeoutRetryPolicy.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/RecoveryCallback.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/RetryStatistics.java
* /opt/cola/permits/1183890210_1627494280.23/0/spring-retry-1-3-1-sources-2-jar/org/springframework/retry/interceptor/MethodInvocationRecoverer.java

1.57 jooq-meta 3.16.4

1.57.1 Available under license :

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Other licenses:

Commercial licenses for this work are available. These replace the above
ASL 2.0 and offer limited warranties, support, maintenance, and commercial
database integrations.

For more information, please visit: http://www.jooq.org/licenses

1.58 apache-commons-lang 3.12.0

1.58.1 Available under license :

Apache Commons Lang
Copyright 2001-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.59 jakarta-activation 1.2.2

1.59.1 Available under license :

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

- Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
  documentation and/or other materials provided with the distribution.

- Neither the name of the Eclipse Foundation, Inc. nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Notices for Jakarta Activation

This content is produced and maintained by Jakarta Activation project.

* Project home: https://projects.eclipse.org/projects/ee4j.jaf

Copyright

All content is the property of the respective authors or their employers. For
more information regarding authorship of content, please consult the listed
source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms
of the Eclipse Distribution License v. 1.0,
which is available at http://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaf

Third-party Content

This project leverages the following third party content.

JUnit (4.12)

* License: Eclipse Public License

1.60 fabric8-::-kubernetes-model-::-scheduling 5.12.1

1.60.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340816092_1654860420.9645436/0/kubernetes-model-scheduling-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-scheduling/pom.xml
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340816092_1654860420.9645436/0/kubernetes-model-scheduling-5-12-1-sources-jar/manifest.vm

1.61 jackson-datatype-jdk8 2.13.3

1.61.1 Available under license :

Apache-2.0

1.62 asm-tree 9.2

1.62.1 Available under license :

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
*

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*   contributors may be used to endorse or promote products derived from
*   this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/package.html
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//   notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//   notice, this list of conditions and the following disclaimer in the
//   documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//   this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/ModuleNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/JumpInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/ModuleProvideNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-

jar/org/objectweb/asm/tree/AnnotationNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/VarInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/TryCatchBlockNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/LocalVariableAnnotationNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/IntInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/InvokeDynamicInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/TypeInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/FieldNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/TypeAnnotationNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/Util.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/InsnList.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/ModuleExportNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/AbstractInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/ClassNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/IincInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/MultiANewArrayInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/LdcInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/InsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/LabelNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/FrameNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/MethodInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/MethodNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/LookupSwitchInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/RecordComponentNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/TableSwitchInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/InnerClassNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/FieldInsnNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-

jar/org/objectweb/asm/tree/ModuleRequireNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/ParameterNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/LocalVariableNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/ModuleOpenNode.java
* /opt/cola/permits/1288695058_1647366277.34/0/asm-tree-9-2-sources-jar/org/objectweb/asm/tree/LineNumberNode.java

1.63 fabric8-::-kubernetes-model-::-autoscaling 5.12.1

1.63.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

       http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340816291_1654860945.9170408/0/kubernetes-model-autoscaling-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-autoscaling/pom.xml
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*

* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340816291_1654860945.9170408/0/kubernetes-model-autoscaling-5-12-1-sources-jar/manifest.vm

1.64 spring-cloud-sleuth-api 3.1.3

1.64.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2013-2018 the original author or authors.
~
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.
-->

Found in path(s):
* /opt/cola/permits/1519136918_1671710591.694181/0/spring-cloud-sleuth-api-3-1-3-jar/META-INF/maven/org.springframework.cloud/spring-cloud-sleuth-api/pom.xml

1.65 bouncy-castle 1.68

1.65.1 Available under license :

Copyright (c) 2000-2021 The Legion of the Bouncy Castle Inc. (https://www.bouncycastle.org)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation,

method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients. rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient.s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as

.Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY.S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item,. as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction.s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys. fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or

regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all

the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the

Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though

third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public

License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception statement from your version.
JUnit

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for

damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that

Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of

such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

1.66 awaitility 4.2.0

1.66.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2022 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
```

* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/TerminalFailureException.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/FailFastCondition.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/Durations.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/DurationFactory.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/TemporalDuration.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/StartEvaluationEvent.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ForeverDuration.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/TimeoutEvent.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/Awaitility.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2008 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/reflect/WhiteboxImpl.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/reflect/exception/FieldNotFoundException.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/FieldSupplierBuilder.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ConditionFactory.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/CallableCondition.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/HamcrestToStringFilter.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/reflect/FieldAnnotationMatcherStrategy.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/reflect/FieldNameMatcherStrategy.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ConditionSettings.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/reflect/FieldMatcherStrategy.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/Condition.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ConditionEvaluationListener.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ConditionEvaluationLogger.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/AbstractHamcrestCondition.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ConditionAwaiter.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/reflect/exception/TooManyFieldsFoundException.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/reflect/FieldTypeMatcherStrategy.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/CallableHamcrestCondition.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2018 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ExecutorLifecycle.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/EvaluationCleanup.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/Uninterruptibles.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2009 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/reflect/FieldNameAndTypeMatcherStrategy.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/reflect/AssignableToFieldTypeMatcherStrategy.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2010 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/spi/Timeout.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2017 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ConditionEvaluationResult.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/classpath/ClassPathResolver.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ExceptionIgnorer.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/EvaluatedCondition.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/DeadlockException.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/pollinterval/PollInterval.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/pollinterval/IterativePollInterval.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ConditionTimeoutException.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ThrowingRunnable.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/PredicateExceptionIgnorer.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/pollinterval/FibonacciPollInterval.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/pollinterval/FixedPollInterval.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ConditionEvaluator.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/HamcrestExceptionIgnorer.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2014 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/ConditionEvaluationHandler.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/LambdaErrorMessageGenerator.java
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/AssertionCondition.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010 PowerMock original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/CheckedExceptionRethrower.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2007 Google Inc.
*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816180_1654719805.9445395/0/awaitility-4-2-0-sources-jar/org/awaitility/core/SameThreadExecutorService.java

1.67 assertj-fluent-assertions 3.22.0

1.67.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link File}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation"
*   for more details.
*
* @author David DIDIER
* @author Yvonne Wang
* @author Alex Ruiz

* @author Olivier Michallat
* @author Olivier Demeijer
* @author Mikhail Mazursky
* @author Jean-Christophe Gay
* @author Valeriy Vyrva
* @author Nikolaos Georgiou
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractFileAssert.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Float}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation"
*   for more details.
*
* @author Drummond Dawson
* @author Yvonne Wang
* @author Alex Ruiz
* @author Ansgar Konermann
* @author Mikhail Mazursky
* @author Nicolas François
* @author Jin Kwon
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractFloatAssert.java

No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@code CharSequence}s.
*
* @param <SELF> the "self" type of this assertion class. Please read &amp;quot;<a href="http://bit.ly/1IZIRcY"
*         target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation</a>&amp;quot;
*         for more details.
* @param <ACTUAL> the type of the "actual" value.
*
* @author Yvonne Wang
* @author David DIDIER
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author Nicolas Francois
* @author Daniel Weber
*/
```

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractCharSequenceAssert.java

No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Short}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation"
*   for more details.
*
* @author Drummond Dawson
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
* @author Alex Ruiz
* @author Mikhail Mazursky
* @author Nicolas François
* @author Cal027
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractShortAssert.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Double}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation"

```
*         for more details.
*
* @author Drummond Dawson
* @author Yvonne Wang
* @author David DIDIER
* @author Alex Ruiz
* @author Ansgar Konermann
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author Nicolas François
* @author Jack Gough
*/
```

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractDoubleAssert.java
No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link List}s.
* @param <SELF> the "self" type of this assertion class. Please read &amp;quot;<a href="http://bit.ly/1IZIRcY"
*         target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation</a>&amp;quot;
*         for more details.
* @param <ACTUAL> the type of the "actual" value.
* @param <ELEMENT> the type of elements of the "actual" value.
* @param <ELEMENT_ASSERT> used for navigational assertions to return the right assert type.
*
* @author Yvonne Wang
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author Jacek Jackowiak
*/
```

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractListAssert.java
No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Byte}s.
*
* @param <SELF> the "self" type of this assertion class. Please read &amp;quot;<a href="http://bit.ly/1IZIRcY"
*        target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation</a>&amp;quot;
*        for more details.
*
* @author Drummond Dawson
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
* @author Alex Ruiz
* @author Mikhail Mazursky
* @author Nicolas François
* @author Cal027
*/
```

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractByteAssert.java
No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
```

* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link InputStream}s.
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation"
*   for more details.
* @param <ACTUAL> the type of the "actual" value.
*
* @author Matthieu Baechler
* @author Mikhail Mazursky
* @author Stefan Birkner
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractInputStreamAssert.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/FutureAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveAllNullFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/recursive/comparison/RecursiveComparisonConfiguration.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/AssertionErrorCreator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Java6Assertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainKey.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/OffsetDateTimeByInstantComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ClassAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotMatch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractCompletableFutureAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ErrorMessageFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/AbstractComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSameClass.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractSpliteratorAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Files.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/FieldComparators.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveBinaryContent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AssertionErrorCollector.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeUpperCase.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldBeExactly.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotHaveSameHashCode.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/description/Description.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Int2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeOdd.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeMarked.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldHaveNoParent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/OptionalDoubleAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/introspection/FieldUtils.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldHaveNoHost.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractIntArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAtSameInstant.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/CommonValidations.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainExactlyInAnyOrder.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AtomicIntegerArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/OptionalLongAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/condition/Join.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveParent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractIteratorAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/extractor/Extractors.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/TextFileWriter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainPattern.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/description/TextDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AtomicLongArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldOnlyHaveFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/DualValueDeque.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/NioFilesWrapper.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/UnsatisfiedRequirement.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldExist.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Boolean2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/internal/IterableElementComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/JUnitBDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Urls.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AssertionInfo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Fail.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/WholeNumbers.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeRelativePath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldHaveQuery.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/BDDAssumptions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/introspection/Introspection.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldHavePath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Maps.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/junit/jupiter/SoftAssertionsExtension.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeEqualComparingFieldByFieldRecursively.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ObjectAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractInstantAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/IterableSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Byte2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeToday.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldSatisfy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/CharacterAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Iterables.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ElementsShouldHaveExactly.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeBlank.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/EnumerableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeInfinite.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/DiffAlgorithm.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainCharSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractUrlAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ComparatorBasedComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/MapSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHavePropertyOrFieldWithValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHavePeriod.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveName.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractPredicateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualWithTimePrecision.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/BigDecimalAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Float2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveDimensions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveOnlyElementsOfType.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/introspection/MethodSupport.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/introspection/PropertySupport.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/BDDAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractDurationAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveNoSuperclass.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/BDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/recursive/comparison/ComparisonKeyDifference.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/InputStreamAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldBeCancelled.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInSameHour.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInfinite.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeUnmodifiable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/UriAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainOnlyDigits.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractOptionalAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Assertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/OptionalDoubleShouldHaveValueCloseToPercentage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAnArray.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotHaveDuplicates.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractEnumerableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/IterableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainSequenceOfCharSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/FloatAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualIgnoringNewLineDifferences.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ArraySortedAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Short2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualIgnoringNewLines.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeFinite.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ZippedElementsShouldSatisfy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Objects.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/BooleanArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldHaveAtLeast.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeReadable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainExactly.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/matcher/AssertionMatcher.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/AbstractShouldHaveTextContent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeNormalized.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeRegularFile.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/configuration/Configuration.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeCanonicalPath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/RecursiveComparisonDifferenceCalculator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ChronoLocalDateTimeComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/ClassNameComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInterface.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/CharArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeBeforeYear.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSameHashCode.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/AssertionErrorMessagesAggregrator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Closeables.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/condition/DoesNotHave.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/DefaultAssertionErrorCollector.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveRootCause.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeInstanceOfAny.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Float2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldNotHave.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/NotThrownAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Booleans.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AssertionsForInterfaceTypes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/condition/AllOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/DurationAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Boolean2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualNormalizingUnicode.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/LongAdderAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSameHourAs.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/array2d/Array2dElementShouldBeDeepEqual.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeBefore.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractBooleanArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/BasicErrorMessageFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/IntArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAbstract.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/WithThrowable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveMessageFindingMatchRegex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInSameHourWindow.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveNext.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/SoftProxies.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualToIgnoringFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveCauseReference.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveRootCauseInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/LocalDateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldBeCompletedExceptionally.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/WithAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInTheFuture.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldHaveFailedWithin.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/filter/InFilter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainCharSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Doubles.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeSame.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEven.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotHaveToString.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainKeys.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Array2DAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/condition/MappedCondition.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/presentation/UnicodeRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContain.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSuppressedException.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/presentation/StandardRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/BigIntegerAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Arrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInSameMonth.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/LocalTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ClassBasedNavigableIterableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/description/JoinDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldNotEndWith.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/DateUtil.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeCloseTo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/data/Offset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/GenericComparableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/SoftAssertionsStatement.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveRootCauseExactlyInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Long2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/DigestDiff.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveExtension.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Uris.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/iterable/ThrowingExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainAnyOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Double2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveAnnotations.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeSubstring.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ListAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/BinaryDiffResult.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/myers/Snake.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveCauseExactlyInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/NoElementsShouldSatisfy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainAnyWhitespaces.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ShortArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualWithinPercentage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/uri/ShouldHaveAnchor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/data/MapEntry.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ErrorCollector.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/InstanceOfAssertFactories.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/IterableUtil.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/DeleteDelta.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/StandardSoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Strings.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractFloatArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/exception/PathsException.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ObjectArrayElementComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSameSizeAs.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/ArrayWrapperList.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AssertDelegateTarget.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/OptionalShouldBePresent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSameDimensionsAs.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractStringAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainOnlyKeys.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/DiffUtils.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractTemporalAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/filter/Filters.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/DoublePredicateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/CharArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldBeSymbolicLink.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ElementsSatisfyingConsumer.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Digests.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractLocalDateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/FieldLocation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveScale.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInSameSecondWindow.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ErrorMessages.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainSubsequenceOfCharSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/NoElementsShouldMatch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualIgnoringHours.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveAtLeastOneElementOfType.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/StandardComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveCause.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainOneOrMoreWhitespaces.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/myers/MyersDiff.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/myers/PathNode.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/FloatComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeBeforeOrEqualTo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractOffsetTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Java6JUnitBDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/junit/jupiter/InjectSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Throwables.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractAtomicFieldUpdaterAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldBeGreater.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBePeriod.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractDoubleArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Dates.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldSatisfy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractPeriodAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInSameMinute.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/introspection/FieldSupport.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/DoubleArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSameTime.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/filter/NotInFilter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHavePropertyOrField.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldNotBeCompletedExceptionally.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/AtomicReferenceArrayElementComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldAccept.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/groups/FieldsOrPropertiesExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/OffsetDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Java6SoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveTime.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractByteArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/SoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveNoExtension.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/BooleanArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/AtomicReferenceAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveNoSuppressedExceptions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Characters.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotStartWith.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/DescribableComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/OptionalShouldContainInstanceOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractBigIntegerAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ByteArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeSubsetOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Char2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Files.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/DeltaComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/CanIgnoreReturnValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/IndexedObjectEnumerableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Lists.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Arrays2D.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualWithinOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractPredicateLikeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSize.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/condition/AnyOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/HamcrestCondition.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldHaveHost.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/TypeComparators.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AtomicLongFieldUpdaterAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldBeAfterOrEqualTo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeIn.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldBeEqualToWithSortedQueryParameters.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/InputStreams.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/LocalDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/FailureMessages.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/description/LazyTextDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInSameMinuteWindow.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldEndWith.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/OptionalShouldContain.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldHavePort.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInstanceOfAny.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/OptionalShouldBeEmpty.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/OptionalIntAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/AnyElementShouldMatch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ClassBasedNavigableListAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AssertionsForClassTypes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/CheckReturnValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ZonedDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Java6BDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AtomicReferenceArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AtomicBooleanAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/SoftAssertionsRule.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/data/TemporalUnitLessThanOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/description/EmptyTextDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Assumptions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/LongArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldEndWithPath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractIterableSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSameContent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Abstract2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ConstructorInvoker.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/WritableAssertionInfo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/presentation/Representation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldNotHave.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEmpty.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveToString.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveDigest.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractOffsetDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/data/TemporalUnitWithinOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAssignableFrom.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldBe.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainOnlyNulls.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Short2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/JUnitSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSizeLessThanOrEqualTo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Comparables.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualNormalizingPunctuationAndWhitespace.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/internal/Lists.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/FactoryBasedNavigableListAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAfter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Byte2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldOnlyHaveElementsOfTypes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ClassModifierShouldBe.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/AbstractComparableNumberComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/ComparisonDifference.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/OptionalAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ChronoZonedDateTimeByInstantComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ByteAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractLocalDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainAtIndex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/SubarraysShouldHaveSameSize.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/SpliteratorAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractCharArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/InsertDelta.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AtomicIntegerAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractZonedDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/FieldMessages.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeLess.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/UnambiguousRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AssertProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContain.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldContainSubsequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/VisibleForTesting.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveDuration.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractObjectArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Predicates.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldBeDone.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/BooleanAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Throwables.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInThePast.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeLowerCase.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotExist.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AutoCloseableSoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Failures.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/condition/Negative.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/TypeHolder.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualIgnoringSeconds.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/TypeMessages.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAtIndex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/ChangeDelta.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/MultipleAssertionsError.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/BinaryDiff.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldBeCompletedWithin.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractLongArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/uri/ShouldHaveParameter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualIgnoringTimezone.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/IteratorAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ThrowableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/IntegerAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Floats.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldBeCompleted.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractFileSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBe.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/junit/jupiter/SoftlyExtension.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/BigIntegers.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractLocalTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/CollectionAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractMapSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/IterableDiff.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/presentation/HexadecimalRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/introspection/ClassUtils.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/MapAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeEmpty.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ShortAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/Chunk.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/annotations/Beta.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ClassLoadingStrategyFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHavePackage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/future/Warning.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Sets.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInSameSecond.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveMethods.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/SoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/DoubleArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Long2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ThrowableAssertAlternative.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ThrowableTypeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldMatch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractAtomicReferenceAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ComparableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/AssertJMultipleFailuresError.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeFinite.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeMixedCase.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ComparatorFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AssertJProxySetup.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractUriAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/condition/Not.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualIgnoringMinutes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Java6JUnitSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldHaveFailed.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/PathAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSuperclass.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHave.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AfterAssertionErrorCollected.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/GroupTypeDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveReference.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/myers/DiffNode.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Maps.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeWritable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AutoCloseableBDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/MessageFormatter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualIgnoringWhitespace.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveMessageMatchingRegex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEmptyDirectory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/AssertionErrorMessagesAggregator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/IntPredicateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/data/TemporalUnitOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AtomicReferenceFieldUpdaterAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAfterYear.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Descriptable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ThrowingConsumer.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldStartWith.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/extractor/ByNameSingleExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/presentation/BinaryRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveStamp.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Java6BDDSoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractComparableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSizeGreaterThanOrEqualTo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveExactlyTypes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/Delta.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/CompletableFutureAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Int2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/data/Index.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ByteArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/presentation/CompositeRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ExtendedByTypesComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Char2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AutoCloseableSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeEqual.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ShortArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/configuration/Services.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainEntries.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeEqualWithinPercentage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Java6BDDAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/DeepDifference.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainPattern.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/PeriodAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ElementsShouldMatch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeExactlyInstanceOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeExhausted.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/PredicateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Double2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractShortArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/FloatArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeNull.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotMatchPattern.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeTrue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldNotHaveFailed.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Preconditions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/OffsetTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/iterable/Extractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainNull.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/IntArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AssertFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AssumptionExceptionFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotAccept.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractOptionalLongAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldMatchPattern.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveNoFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/BigIntegerComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Arrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/InstanceOfAssertFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/FloatingPointNumberAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/presentation/NumberGrouping.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Condition.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeEqualNormalizingWhitespace.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeFile.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ConditionAndGroupGenericParameterTypeShouldBeTheSame.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeEqualIgnoringWhitespace.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInSameYear.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/myers/Equalizer.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/groups/Tuple.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeBlank.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeSorted.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AtomicLongAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/FieldByFieldComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/RealNumbers.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveLineCount.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAbsolutePath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/NaturalOrderComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualByComparingFieldByFieldRecursively.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBe.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ConfigurableRecursiveFieldByFieldComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/DoubleComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Paths.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/internal/Longs.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/filter/NotFilter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainCharSequenceOnlyOnce.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/FileSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/SoftAssertionError.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractOptionalDoubleAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/FieldHolder.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeOfClassIn.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeBase64.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/condition/VerboseCondition.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Object2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSizeLessThan.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldBeAtMost.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/BDDSoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ObjectArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/BigDecimals.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainValues.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/TriFunction.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldNotBeCancelled.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Diff.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Shorts.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ExtensionPoints.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/data/TemporalOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldNotBeExactlyInstanceOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/OptionalDoubleShouldHaveValueCloseToOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/OnFieldsComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeBetween.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldHaveFragment.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/FactoryBasedNavigableIterableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/DateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/introspection/MemberUtils.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSizeGreaterThan.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Streams.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/NullSafeComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/LongArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Conditions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/StringAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAnnotation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/LongAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/DoubleAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveContent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeIn.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Spliterators.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualByComparingOnlyGivenFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/PathNaturalOrderComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Hexadecimals.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Futures.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldContainRecursively.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainKeys.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeGreaterOrEqual.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldNotBeCompleted.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/InputStreamsException.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/introspection/IntrospectionError.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Java6StandardSoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/DualValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AtomicIntegerFieldUpdaterAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/CommonErrors.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/JUnitJupiterBDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldBeAtLeast.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainEntry.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/FileAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AtomicStampedReferenceAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeLessOrEqual.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeNullOrEmpty.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveCauseInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Assert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotHaveThrown.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqual.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/groups/Properties.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/UrlAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainsOnlyOnce.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/AtomicMarkableReferenceAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Classes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/InstantAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualIgnoringNanos.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/xml/XmlStringPrettyFormatter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/configuration/PreferredAssumptionException.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainNull.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/data/Percentage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/extractor/ResultOfExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Iterators.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/WithAssumptions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/AssertionErrorFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/LongPredicateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldNotBe.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainSubsequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualNormalizingWhitespace.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainOnlyWhitespaces.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/JUnitJupiterSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainOnlyWhitespaces.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/BigDecimalComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/configuration/ConfigurationProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractFutureAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/CharSequenceAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveMessage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/ProxifyMethodChangingTheObjectUnderTest.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldHaveScheme.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Bytes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractPathAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/Objects.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldHave.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Object2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeDirectory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInSameDay.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Numbers.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSizeBetween.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldHaveAuthority.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/introspection/PropertyOrFieldSupport.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ObjectArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainAtIndex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveDateField.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeFalse.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeExecutable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeSame.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/future/ShouldNotBeDone.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeBetween.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/NumberAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveNoCause.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotHaveAnyElementsOfTypes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/util/diff/Patch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractOptionalIntAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeEqualIgnoringCase.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Strings.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/filter/FilterOperator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainKey.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldHaveUserInfo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/extractor/ToStringExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/IgnoringFieldsComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeEqualWithinOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveAtIndex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/presentation/PredicateDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/extractor/ByNameMultipleExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotHaveSameClass.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainOnly.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotBeOfClassIn.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/DescriptionFormatter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Integers.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualIgnoringCase.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/FloatArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ObjectAssertFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveNoNullFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/URLs.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/uri/ShouldHaveProtocol.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldStartWithPath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/RecursiveComparisonAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldHaveAtMost.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all assertions.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation"
*   for more details.
* @param <ACTUAL> the type of the "actual" value.
*
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author Nicolas François
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractAssert.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*

* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Boolean}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation"
*   for more details.
*
* @author Alex Ruiz
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
* @author Mikhail Mazursky
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractBooleanAssert.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Object}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API

implementation"
* for more details.
* @param <ACTUAL> the type of the "actual" value.
*
* @author Yvonne Wang
* @author Alex Ruiz
* @author Nicolas François
* @author Mikhail Mazursky
* @author Joel Costigliola
* @author Libor Ondrusek
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractObjectAssert.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Assertions methods applicable to groups of objects (e.g. arrays or collections.)
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
* target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation"
* for more details.
* @param <ELEMENT> the type of elements of the "actual" value.
*
* @author Yvonne Wang
* @author Alex Ruiz
* @author Nicolas François
* @author Mikhail Mazursky
* @author Joel Costigliola
* @author Nicolas François
* @author Florent Biville
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ObjectEnumerableAssert.java
No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Date}s.
* <p>
* Note that assertions with date parameter comes with two flavor, one is obviously a {@link Date} and the other is a
* {@link String} representing a Date.<br>
* For the latter, the default format follows ISO 8901 : "yyyy-MM-dd", user can override it with a custom format by
* calling {@link #withDateFormat(DateFormat)}.<br>
* The user custom format will then be used for all next Date assertions (i.e not limited to the current assertion) in
* the test suite.<br>
* To turn back to default format, simply call {@link #withDefaultDateFormatsOnly()}.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*        target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation</a>" for
*        more details.
* @author Tomasz Nurkiewicz (thanks for giving assertions idea)
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author William Delanoue
* @author Michal Kordas
* @author Eddú Meléndez
*/
```

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractDateAssert.java
No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Integer}s.
*
* @param <SELF> the "self" type of this assertion class. Please read &amp;quot;<a href="http://bit.ly/1IZIRcY"
*        target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation</a>&amp;quot;
*        for more details.
*
* @author Drummond Dawson
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author Nicolas François
* @author Cal027
*/
```

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractIntegerAssert.java
No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
```

*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Map}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation"
*   for more details.
* @param <ACTUAL> the type of the "actual" value.
* @param <K> the type of keys in the map.
* @param <V> the type of values in the map.
*
* @author David DIDIER
* @author Yvonne Wang
* @author Alex Ruiz
* @author Mikhail Mazursky
* @author Nicolas François
* @author dorzey
* @author Filip Hrisafov
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractMapAssert.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Collection}s.
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation"
*   for more details.

* @param <ACTUAL> the type of the "actual" value.
* @param <ELEMENT> the type of elements of the "actual" value.
* @param <ELEMENT_ASSERT> used for navigational assertions to return the right assert type.
*
* @since 3.21.0
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractCollectionAssert.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Throwable}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation"
*   for more details.
* @param <ACTUAL> the type of the "actual" value.
*
* @author David DIDIER
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author Jack Gough
* @author Mike Gilchrist
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractThrowableAssert.java
No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Class}es.
*
* @param <SELF> the "self" type of this assertion class. Please read &amp;quot;<a href="http://bit.ly/1IZIRcY"
*         target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation</a>&amp;quot;
*         for more details.
*
* @author William Delanoue
* @author Mikhail Mazursky
*/
```

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractClassAssert.java
No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for implementations of <code>{@link ObjectEnumerableAssert}</code> whose actual value type is
* <code>{@link Collection}</code>.
```

```
*
* @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
*        target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation</a>&quot;
*        for more details.
* @param <ACTUAL> the type of the "actual" value.
* @param <ELEMENT> the type of elements of the "actual" value.
* @param <ELEMENT_ASSERT> used for navigational assertions to return the right assert type.
*
* @author Yvonne Wang
* @author Alex Ruiz
* @author Mathieu Baechler
* @author Joel Costigliola
* @author Maciej Jaskowski
* @author Nicolas François
* @author Mikhail Mazursky
* @author Mateusz Haligowski
* @author Lovro Pandzic
* @author Marko Bekhta
*/
```

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractIterableAssert.java
No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link BigDecimal}s.
*
* @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
*        target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation</a>&quot;
*        for more details.
*
```

* @author Drummond Dawson
* @author David DIDIER
* @author Ted M. Young
* @author Yvonne Wang
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractBigDecimalAssert.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Character}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation"
*   for more details.
*
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractCharacterAssert.java

No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link Long}s.
*
* @param <SELF> the "self" type of this assertion class. Please read &amp;quot;<a href="http://bit.ly/1IZIRcY"
*        target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation</a>&amp;quot;
*        for more details.
*
* @author Drummond Dawson
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author Nicolas François
* @author Cal027
*/
```

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractLongAssert.java

No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
```

* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for {@link LongAdder}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
*   target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation"
*   for more details.
*
* @author Grzegorz Piwowarek
* @since 3.16.0
*/

Found in path(s):
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractLongAdderAssert.java

1.68 commons-compress 1.21

1.68.1 Available under license :

Apache Commons Compress
Copyright 2002-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

The files in the package org.apache.commons.compress.archivers.sevenz
were derived from the LZMA SDK, version 9.20 (C/ and CPP/7zip/),
which has been placed in the public domain:

"LZMA SDK is placed in the public domain." (http://www.7-zip.org/sdk.html)

The test file lbzip2_32767.bz2 has been copied from libbzip2's source
repository:

This program, "bzip2", the associated library "libbzip2", and all
documentation, are copyright (C) 1996-2019 Julian R Seward. All
rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must
not claim that you wrote the original software. If you use this
software in a product, an acknowledgment in the product
documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must
not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote
products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS
OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the

Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.69 saxon-xslt-and-xquery-processor 10.6

1.69.1 Available under license :

No license file was found, but licenses were detected in source scan.

XHTML 1.1 plus MathML 2.0 plus SVG 1.1 DTD

This is a prototype extension of XHTML 1.1 incorporating MathML 2.0
and SVG 1.1.

Copyright 2002 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute this DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Editors: Murray M. Altheim <altheim@eng.sun.com> (XHTML modules)
David Carlisle <davidc@nag.co.uk> (MathML modules)
Jun Fujisawa <fujisawa.jun@canon.co.jp> (SVG modules)
Masayasu Ishikawa <mimasa@w3.org> (DTD driver)
Revision:
$Id: xhtml
2001 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the XHTML DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Murray M. Altheim <altheim@eng.sun.com>
Revision: $Id: xhtml
2003 W3C® (MIT, ERCIM, Keio), All Rights
Reserved. W3C liability, trademark, document use and software
licensing rules apply.

Permission to use, copy, modify and distribute the MathML 2.0 DTD and
its accompanying documentation for any purpose and without fee is

hereby granted in perpetuity, provided that the above copyright notice
and this paragraph appear in all copies. The copyright holders make
no representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Revision: $Id: xhtml

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml2/xhtml-math11-f.dtd
No license file was found, but licenses were detected in source scan.

<!--
File isopub.ent produced by the XSL script entities.xsl
from input data in unicode.xml.

Copyright 1998 - 2009 W3C.

Use and distribution of this code are permitted under the terms of the
W3C Software Notice and License.
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

Please report any errors to David Carlisle
via the public W3C list www-math@w3.org.

The numeric character values assigned to each entity
(should) match the Unicode assignments in Unicode 5.x.

Entity names in this file are derived from files carrying the
following notice:

(C) International Organization for Standardization 1986
Permission to copy in any form is granted for use with
conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Publishing//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isopub.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isopub PUBLIC
"-//W3C//ENTITIES Publishing//EN//XML"
"http://www.w3.org/2003/entities/2007/isopub.ent"
>
%isopub;

-->

<!ENTITY blank "␣" ><!--OPEN BOX -->
<!ENTITY blk12 "▒" ><!--MEDIUM SHADE -->
<!ENTITY blk14 "░" ><!--LIGHT SHADE -->
<!ENTITY blk34 "▓" ><!--DARK SHADE -->
<!ENTITY block "█" ><!--FULL BLOCK -->
<!ENTITY bull "•" ><!--BULLET -->
<!ENTITY caret "⁁" ><!--CARET INSERTION POINT -->
<!ENTITY check "✓" ><!--CHECK MARK -->
<!ENTITY cir "○" ><!--WHITE CIRCLE -->
<!ENTITY clubs "♣" ><!--BLACK CLUB SUIT -->
<!ENTITY copysr "℗" ><!--SOUND RECORDING COPYRIGHT -->
<!ENTITY cross "✗" ><!--BALLOT X -->
<!ENTITY dagger "†" ><!--DAGGER -->
<!ENTITY Dagger "‡" ><!--DOUBLE DAGGER -->
<!ENTITY dash "‐" ><!--HYPHEN -->
<!ENTITY diams "♦" ><!--BLACK DIAMOND SUIT -->
<!ENTITY dlcrop "⌍" ><!--BOTTOM LEFT CROP -->
<!ENTITY drcrop "⌌" ><!--BOTTOM RIGHT CROP -->
<!ENTITY dtri "▿" ><!--WHITE DOWN-POINTING SMALL TRIANGLE -->
<!ENTITY dtrif "▾" ><!--BLACK DOWN-POINTING SMALL TRIANGLE -->
<!ENTITY emsp " " ><!--EM SPACE -->
<!ENTITY emsp13 " " ><!--THREE-PER-EM SPACE -->
<!ENTITY emsp14 " " ><!--FOUR-PER-EM SPACE -->
<!ENTITY ensp " " ><!--EN SPACE -->
<!ENTITY female "♀" ><!--FEMALE SIGN -->
<!ENTITY ffilig "ﬃ" ><!--LATIN SMALL LIGATURE FFI -->
<!ENTITY fflig "ﬀ" ><!--LATIN SMALL LIGATURE FF -->
<!ENTITY ffllig "ﬄ" ><!--LATIN SMALL LIGATURE FFL -->
<!ENTITY filig "ﬁ" ><!--LATIN SMALL LIGATURE FI -->
<!ENTITY fjlig "fj" ><!--fj ligature -->
<!ENTITY flat "♭" ><!--MUSIC FLAT SIGN -->
<!ENTITY fllig "ﬂ" ><!--LATIN SMALL LIGATURE FL -->
<!ENTITY frac13 "⅓" ><!--VULGAR FRACTION ONE THIRD -->
<!ENTITY frac15 "⅕" ><!--VULGAR FRACTION ONE FIFTH -->
<!ENTITY frac16 "⅙" ><!--VULGAR FRACTION ONE SIXTH -->
<!ENTITY frac23 "⅔" ><!--VULGAR FRACTION TWO THIRDS -->
<!ENTITY frac25 "⅖" ><!--VULGAR FRACTION TWO FIFTHS -->
<!ENTITY frac35 "⅗" ><!--VULGAR FRACTION THREE FIFTHS -->

```
<!ENTITY frac45         "&#x02158;" ><!--VULGAR FRACTION FOUR FIFTHS -->
<!ENTITY frac56         "&#x0215A;" ><!--VULGAR FRACTION FIVE SIXTHS -->
<!ENTITY hairsp         " " ><!--HAIR SPACE -->
<!ENTITY hearts         "&#x02665;" ><!--BLACK HEART SUIT -->
<!ENTITY hellip         "&#x02026;" ><!--HORIZONTAL ELLIPSIS -->
<!ENTITY hybull         "&#x02043;" ><!--HYPHEN BULLET -->
<!ENTITY incare         "&#x02105;" ><!--CARE OF -->
<!ENTITY ldquor         "&#x0201E;" ><!--DOUBLE LOW-9 QUOTATION MARK -->
<!ENTITY lhblk          "&#x02584;" ><!--LOWER HALF BLOCK -->
<!ENTITY loz            "&#x025CA;" ><!--LOZENGE -->
<!ENTITY lozf           "&#x029EB;" ><!--BLACK LOZENGE -->
<!ENTITY lsquor         "&#x0201A;" ><!--SINGLE LOW-9 QUOTATION MARK -->
<!ENTITY ltri          "&#x025C3;" ><!--WHITE LEFT-POINTING SMALL TRIANGLE -->
<!ENTITY ltrif         "&#x025C2;" ><!--BLACK LEFT-POINTING SMALL TRIANGLE -->
<!ENTITY male           "&#x02642;" ><!--MALE SIGN -->
<!ENTITY malt           "&#x02720;" ><!--MALTESE CROSS -->
<!ENTITY marker          "&#x025AE;" ><!--BLACK VERTICAL RECTANGLE -->
<!ENTITY mdash          "&#x02014;" ><!--EM DASH -->
<!ENTITY mldr           "&#x02026;" ><!--HORIZONTAL ELLIPSIS -->
<!ENTITY natur          "&#x0266E;" ><!--MUSIC NATURAL SIGN -->
<!ENTITY ndash          "&#x02013;" ><!--EN DASH -->
<!ENTITY nldr           "&#x02025;" ><!--TWO DOT LEADER -->
<!ENTITY numsp          " " ><!--FIGURE SPACE -->
<!ENTITY phone          "&#x0260E;" ><!--BLACK TELEPHONE -->
<!ENTITY puncsp         " " ><!--PUNCTUATION SPACE -->
<!ENTITY rdquor         "&#x0201D;" ><!--RIGHT DOUBLE QUOTATION MARK -->
<!ENTITY rect           "&#x025AD;" ><!--WHITE RECTANGLE -->
<!ENTITY rsquor         "&#x02019;" ><!--RIGHT SINGLE QUOTATION MARK -->
<!ENTITY rtri          "&#x025B9;" ><!--WHITE RIGHT-POINTING SMALL TRIANGLE -->
<!ENTITY rtrif         "&#x025B8;" ><!--BLACK RIGHT-POINTING SMALL TRIANGLE -->
<!ENTITY rx            "&#x0211E;" ><!--PRESCRIPTION TAKE -->
<!ENTITY sext           "&#x02736;" ><!--SIX POINTED BLACK STAR -->
<!ENTITY sharp          "&#x0266F;" ><!--MUSIC SHARP SIGN -->
<!ENTITY spades         "&#x02660;" ><!--BLACK SPADE SUIT -->
<!ENTITY squ            "&#x025A1;" ><!--WHITE SQUARE -->
<!ENTITY squf           "&#x025AA;" ><!--BLACK SMALL SQUARE -->
<!ENTITY star           "&#x02606;" ><!--WHITE STAR -->
<!ENTITY starf          "&#x02605;" ><!--BLACK STAR -->
<!ENTITY target         "&#x02316;" ><!--POSITION INDICATOR -->
<!ENTITY telrec         "&#x02315;" ><!--TELEPHONE RECORDER -->
<!ENTITY thinsp         " " ><!--THIN SPACE -->
<!ENTITY uhblk          "&#x02580;" ><!--UPPER HALF BLOCK -->
<!ENTITY ulcrop         "&#x0230F;" ><!--TOP LEFT CROP -->
<!ENTITY urcrop         "&#x0230E;" ><!--TOP RIGHT CROP -->
<!ENTITY utri           "&#x025B5;" ><!--WHITE UP-POINTING SMALL TRIANGLE -->
<!ENTITY utrif          "&#x025B4;" ><!--BLACK UP-POINTING SMALL TRIANGLE -->
<!ENTITY vellip         "&#x022EE;" ><!--VERTICAL ELLIPSIS -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isopub.ent
No license file was found, but licenses were detected in source scan.

2007 World Wide Web Consortium
(Massachusetts Institute of Technology, European Research Consortium
for Informatics and Mathematics, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the XHTML Basic DTD
and its accompanying documentation for any purpose and without fee is
hereby granted in perpetuity, provided that the above copyright notice
and this paragraph appear in all copies. The copyright holders make
no representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Editors: Murray M. Altheim <mailto:altheim@eng.sun.com>
Peter Stark <mailto:Peter.Stark@ecs.ericsson.se>
Shane McCarron <mailto:shane@aptest.com>
Revision: $Id: xhtml

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-basic11.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Datatypes Module ..... -->
<!-- file: svg-datatypes.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-datatypes.mod,v 1.2 2002/04/20 18:07:42 fujisawa Exp $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Datatypes//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-datatypes.mod"

..... -->

<!-- Datatypes

This module declares common data types for properties and attributes.
-->

```
<!-- feature specification -->
<!ENTITY % Boolean.datatype "( false | true )" >

<!-- 'clip-rule' or 'fill-rule' property/attribute value -->
<!ENTITY % ClipFillRule.datatype "( nonzero | evenodd | inherit )" >

<!-- media type, as per [RFC2045] -->
<!ENTITY % ContentType.datatype "CDATA" >

<!-- a <coordinate> -->
<!ENTITY % Coordinate.datatype "CDATA" >

<!-- a list of <coordinate>s -->
<!ENTITY % Coordinates.datatype "CDATA" >

<!-- a <color> value -->
<!ENTITY % Color.datatype "CDATA" >

<!-- a <integer> -->
<!ENTITY % Integer.datatype "CDATA" >

<!-- a language code, as per [RFC3066] -->
<!ENTITY % LanguageCode.datatype "NMTOKEN" >

<!-- comma-separated list of language codes, as per [RFC3066] -->
<!ENTITY % LanguageCodes.datatype "CDATA" >

<!-- a <length> -->
<!ENTITY % Length.datatype "CDATA" >

<!-- a list of <length>s -->
<!ENTITY % Lengths.datatype "CDATA" >

<!-- a <number> -->
<!ENTITY % Number.datatype "CDATA" >

<!-- a list of <number>s -->
<!ENTITY % Numbers.datatype "CDATA" >

<!-- opacity value (e.g., <number>) -->
<!ENTITY % OpacityValue.datatype "CDATA" >

<!-- a path data specification -->
<!ENTITY % PathData.datatype "CDATA" >

<!-- 'preserveAspectRatio' attribute specification -->
<!ENTITY % PreserveAspectRatioSpec.datatype "CDATA" >
```

<!-- script expression -->
<!ENTITY % Script.datatype "CDATA" >

<!-- An SVG color value (RGB plus optional ICC) -->
<!ENTITY % SVGColor.datatype "CDATA" >

<!-- arbitrary text string -->
<!ENTITY % Text.datatype "CDATA" >

<!-- list of transforms -->
<!ENTITY % TransformList.datatype "CDATA" >

<!-- a Uniform Resource Identifier, see [URI] -->
<!ENTITY % URI.datatype "CDATA" >

<!-- 'viewBox' attribute specification -->
<!ENTITY % ViewBoxSpec.datatype "CDATA" >

<!-- end of svg-datatypes.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-datatypes.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Client-side Image Map Module ..... -->
<!-- file: xhtml-csismap-1.mod

This is XHTML, a reformulation of HTML as a modular XML application.
Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
Revision: $Id: xhtml-csismap-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS XHTML Client-side Image Maps 1.0//EN"
SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-csismap-1.mod"

Revisions:
(none)
..... -->

<!-- Client-side Image Maps

area, map

This module declares elements and attributes to support client-side
image maps. This requires that the Image Module (or a module

```
   declaring the img element type) be included in the DTD.

   These can be placed in the same document or grouped in a
   separate document, although the latter isn't widely supported
-->

<!ENTITY % area.element  "INCLUDE" >
<![%area.element;[
<!ENTITY % area.content  "EMPTY" >
<!ENTITY % area.qname  "area" >
<!ELEMENT %area.qname;  %area.content; >
<!-- end of area.element -->]]>

<!ENTITY % Shape.datatype "( rect | circle | poly | default )">
<!ENTITY % Coords.datatype "CDATA" >

<!ENTITY % area.attlist  "INCLUDE" >
<![%area.attlist;[
<!ATTLIST %area.qname;
    %Common.attrib;
    href        %URI.datatype;          #IMPLIED
    shape       %Shape.datatype;        'rect'
    coords      %Coords.datatype;       #IMPLIED
    nohref      ( nohref )              #IMPLIED
    alt         %Text.datatype;         #REQUIRED
    tabindex    %Number.datatype;       #IMPLIED
    accesskey   %Character.datatype;    #IMPLIED
>
<!-- end of area.attlist -->]]>

<!-- modify anchor attribute definition list
   to allow for client-side image maps
-->
<!ATTLIST %a.qname;
    shape       %Shape.datatype;        'rect'
    coords      %Coords.datatype;       #IMPLIED
>

<!-- modify img attribute definition list
   to allow for client-side image maps
-->
<!ATTLIST %img.qname;
    usemap      IDREF                   #IMPLIED
>

<!-- modify form input attribute definition list
   to allow for client-side image maps
-->
```

```
<!ATTLIST %input.qname;
    usemap      IDREF                #IMPLIED
>

<!-- modify object attribute definition list
   to allow for client-side image maps
-->
<!ATTLIST %object.qname;
    usemap      IDREF                #IMPLIED
>

<!-- 'usemap' points to the 'id' attribute of a <map> element,
   which must be in the same document; support for external
   document maps was not widely supported in HTML and is
   eliminated in XHTML.

   It is considered an error for the element pointed to by
   a usemap IDREF to occur in anything but a <map> element.
-->

<!ENTITY % map.element  "INCLUDE" >
<![%map.element;[
<!ENTITY % map.content
   "(( %Block.mix; ) | %area.qname; )+"
>
<!ENTITY % map.qname  "map" >
<!ELEMENT %map.qname;  %map.content; >
<!-- end of map.element -->]]>

<!ENTITY % map.attlist  "INCLUDE" >
<![%map.attlist;[
<!ATTLIST %map.qname;
    %XHTML.xmlns.attrib;
    id          ID                   #REQUIRED
    %class.attrib;
    %title.attrib;
    %Core.extra.attrib;
    %I18n.attrib;
    %Events.attrib;
>
<!-- end of map.attlist -->]]>

<!-- end of xhtml-csismap-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-csismap-1.mod
```

No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Attribute Collection Module ..................................... -->
<!-- file: svg11-attribs.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg11-attribs.mod,v 1.4 2002/11/14 15:11:03 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Attribute Collection//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11-attribs.mod"

   ..................................................................... -->

<!-- SVG 1.1 Attribute Collection

   This module defines the set of common attributes that can be present
   on many SVG elements.
-->

<!-- module: svg-conditional.mod ....................... -->

<!ENTITY % ExtensionList.datatype "CDATA" >
<!ENTITY % FeatureList.datatype "CDATA" >

<!ENTITY % SVG.Conditional.extra.attrib "" >
<!ENTITY % SVG.Conditional.attrib
  "requiredFeatures %FeatureList.datatype; #IMPLIED
   requiredExtensions %ExtensionList.datatype; #IMPLIED
   systemLanguage %LanguageCodes.datatype; #IMPLIED
   %SVG.Conditional.extra.attrib;"
>

<!-- module: svg-style.mod ............................. -->

<!ENTITY % ClassList.datatype "CDATA" >
<!ENTITY % StyleSheet.datatype "CDATA" >

<!ENTITY % SVG.Style.extra.attrib "" >
<!ENTITY % SVG.Style.attrib
  "style %StyleSheet.datatype; #IMPLIED
   class %ClassList.datatype; #IMPLIED
   %SVG.Style.extra.attrib;"
>
```

```
<!-- module: svg-text.mod ..... -->

<!ENTITY % BaselineShiftValue.datatype "CDATA" >
<!ENTITY % FontFamilyValue.datatype "CDATA" >
<!ENTITY % FontSizeValue.datatype "CDATA" >
<!ENTITY % FontSizeAdjustValue.datatype "CDATA" >
<!ENTITY % GlyphOrientationHorizontalValue.datatype "CDATA" >
<!ENTITY % GlyphOrientationVerticalValue.datatype "CDATA" >
<!ENTITY % KerningValue.datatype "CDATA" >
<!ENTITY % SpacingValue.datatype "CDATA" >
<!ENTITY % TextDecorationValue.datatype "CDATA" >

<!ENTITY % SVG.Text.extra.attrib "" >
<!ENTITY % SVG.Text.attrib
  "writing-mode ( lr-tb | rl-tb | tb-rl | lr | rl | tb | inherit ) #IMPLIED
   %SVG.Text.extra.attrib;"
>

<!ENTITY % SVG.TextContent.extra.attrib "" >
<!ENTITY % SVG.TextContent.attrib
  "alignment-baseline ( auto | baseline | before-edge | text-before-edge |
                        middle | central | after-edge | text-after-edge |
                        ideographic | alphabetic | hanging | mathematical |
                        inherit ) #IMPLIED
   baseline-shift %BaselineShiftValue.datatype; #IMPLIED
   direction ( ltr | rtl | inherit ) #IMPLIED
   dominant-baseline ( auto | use-script | no-change | reset-size |
                       ideographic | alphabetic | hanging | mathematical |
                       central | middle | text-after-edge | text-before-edge |
                       inherit ) #IMPLIED
   glyph-orientation-horizontal %GlyphOrientationHorizontalValue.datatype;
                                #IMPLIED
   glyph-orientation-vertical %GlyphOrientationVerticalValue.datatype;
                              #IMPLIED
   kerning %KerningValue.datatype; #IMPLIED
   letter-spacing %SpacingValue.datatype; #IMPLIED
   text-anchor ( start | middle | end | inherit ) #IMPLIED
   text-decoration %TextDecorationValue.datatype; #IMPLIED
   unicode-bidi ( normal | embed | bidi-override | inherit ) #IMPLIED
   word-spacing %SpacingValue.datatype; #IMPLIED
   %SVG.TextContent.extra.attrib;"
>

<!ENTITY % SVG.Font.extra.attrib "" >
<!ENTITY % SVG.Font.attrib
  "font-family %FontFamilyValue.datatype; #IMPLIED
   font-size %FontSizeValue.datatype; #IMPLIED
   font-size-adjust %FontSizeAdjustValue.datatype; #IMPLIED
```

```
    font-stretch ( normal | wider | narrower | ultra-condensed |
                   extra-condensed | condensed | semi-condensed |
                   semi-expanded | expanded | extra-expanded |
                   ultra-expanded | inherit ) #IMPLIED
    font-style ( normal | italic | oblique | inherit ) #IMPLIED
    font-variant ( normal | small-caps | inherit ) #IMPLIED
    font-weight ( normal | bold | bolder | lighter | 100 | 200 | 300 | 400 |
                  500 | 600 | 700 | 800 | 900 | inherit ) #IMPLIED
    %SVG.Font.extra.attrib;"
>

<!-- module: svg-marker.mod ..... -->

<!ENTITY % MarkerValue.datatype "CDATA" >

<!ENTITY % SVG.Marker.extra.attrib "" >
<!ENTITY % SVG.Marker.attrib
   "marker-start %MarkerValue.datatype; #IMPLIED
    marker-mid %MarkerValue.datatype; #IMPLIED
    marker-end %MarkerValue.datatype; #IMPLIED
    %SVG.Marker.extra.attrib;"
>

<!-- module: svg-profile.mod ..... -->

<!ENTITY % SVG.ColorProfile.extra.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib
   "color-profile CDATA #IMPLIED
    %SVG.ColorProfile.extra.attrib;"
>

<!-- module: svg-gradient.mod ..... -->

<!ENTITY % NumberOrPercentage.datatype "CDATA" >

<!ENTITY % SVG.Gradient.extra.attrib "" >
<!ENTITY % SVG.Gradient.attrib
   "stop-color %SVGColor.datatype; #IMPLIED
    stop-opacity %OpacityValue.datatype; #IMPLIED
    %SVG.Gradient.extra.attrib;"
>

<!-- module: svg-clip.mod ..... -->

<!ENTITY % ClipPathValue.datatype "CDATA" >

<!ENTITY % SVG.Clip.extra.attrib "" >
<!ENTITY % SVG.Clip.attrib
```

```
   "clip-path %ClipPathValue.datatype; #IMPLIED
    clip-rule %ClipFillRule.datatype; #IMPLIED
    %SVG.Clip.extra.attrib;"
>

<!-- module: svg-mask.mod ..... -->

<!ENTITY % MaskValue.datatype "CDATA" >

<!ENTITY % SVG.Mask.extra.attrib "" >
<!ENTITY % SVG.Mask.attrib
   "mask %MaskValue.datatype; #IMPLIED
    %SVG.Mask.extra.attrib;"
>

<!-- module: svg-filter.mod ..... -->

<!ENTITY % FilterValue.datatype "CDATA" >
<!ENTITY % NumberOptionalNumber.datatype "CDATA" >

<!ENTITY % SVG.Filter.extra.attrib "" >
<!ENTITY % SVG.Filter.attrib
   "filter %FilterValue.datatype; #IMPLIED
    %SVG.Filter.extra.attrib;"
>

<!ENTITY % SVG.FilterColor.extra.attrib "" >
<!ENTITY % SVG.FilterColor.attrib
   "color-interpolation-filters ( auto | sRGB | linearRGB | inherit )
                                #IMPLIED
    %SVG.FilterColor.extra.attrib;"
>

<!-- module: svg-cursor.mod ..... -->

<!ENTITY % CursorValue.datatype "CDATA" >

<!ENTITY % SVG.Cursor.extra.attrib "" >
<!ENTITY % SVG.Cursor.attrib
   "cursor %CursorValue.datatype; #IMPLIED
    %SVG.Cursor.extra.attrib;"
>

<!-- end of svg11-attribs.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg11-attribs.mod
```

No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 View Module ..................................................... -->
<!-- file: svg-view.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-view.mod,v 1.3 2002/10/24 17:40:16 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 View//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-view.mod"

   ..................................................................... -->

<!-- View

     view

   This module declares markup to provide support for view.
-->

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.view.qname "view" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.View.class ..... -->

<!ENTITY % SVG.View.extra.class "" >

<!ENTITY % SVG.View.class
  "| %SVG.view.qname; %SVG.View.extra.class;"
>

<!-- view: View Element ..... -->

<!ENTITY % SVG.view.extra.content "" >

<!ENTITY % SVG.view.element "INCLUDE" >
<![%SVG.view.element;[
<!ENTITY % SVG.view.content
```

```
  "( %SVG.Description.class; %SVG.view.extra.content; )*"
>
<!ELEMENT %SVG.view.qname; %SVG.view.content; >
<!-- end of SVG.view.element -->]]>

<!ENTITY % SVG.view.attlist "INCLUDE" >
<![%SVG.view.attlist;[
<!ATTLIST %SVG.view.qname;
  %SVG.Core.attrib;
  %SVG.External.attrib;
  viewBox %ViewBoxSpec.datatype; #IMPLIED
  preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
  zoomAndPan ( disable | magnify ) 'magnify'
  viewTarget CDATA #IMPLIED
>
<!-- end of SVG.view.attlist -->]]>

<!-- end of svg-view.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-view.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- XHTML Base Architecture Module  ..................................... -->
<!-- file: xhtml-arch-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-arch-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Base Architecture 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-arch-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- This optional module includes declarations that enable XHTML to be used
   as a base architecture according to the 'Architectural Forms Definition
   Requirements' (Annex A.3, ISO/IEC 10744, 2nd edition). For more information
   on use of architectural forms, see the HyTime web site at:

       http://www.hytime.org/
-->
```

```
<?IS10744 ArcBase xhtml ?>

<!NOTATION xhtml PUBLIC "-//W3C//NOTATION AFDR ARCBASE XHTML 1.1//EN" >

<!-- Entity declaration for associated Architectural DTD
-->
<!ENTITY xhtml-arch.dtd
    PUBLIC "-//W3C//DTD XHTML Architecture 1.1//EN"
         "xhtml11-arch.dtd" >

<?IS10744:arch xhtml
  public-id      =  "-//W3C//NOTATION AFDR ARCBASE XHTML 1.1//EN"
  dtd-public-id   =  "-//W3C//DTD XHTML 1.1//EN"
  dtd-system-id   =  "xhtml11.dtd"
  doc-elem-form   =  "html"
  form-att        =  "html"
  renamer-att     =  "htnames"
  suppressor-att  =  "htsupp"
  data-ignore-att =  "htign"
  auto            =  "ArcAuto"
  options         =  "HtModReq HtModOpt"
  HtModReq        =  "Framework Text Hypertext Lists Structure"
  HtModOpt        =  "Standard"
?>

<!-- end of xhtml-arch-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-arch-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Base Element Module  ............................................ -->
<!-- file: xhtml-base-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-base-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Base Element 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-base-1.mod"

   Revisions:
   (none)
```

```
    ..... -->

<!-- Base element

      base

    This module declares the base element type and its attributes,
    used to define a base URI against which relative URIs in the
    document will be resolved.

    Note that this module also redeclares the content model for
    the head element to include the base element.
-->

<!-- base: Document Base URI ..... -->

<!ENTITY % base.element  "INCLUDE" >
<![%base.element;[
<!ENTITY % base.content  "EMPTY" >
<!ENTITY % base.qname  "base" >
<!ELEMENT %base.qname;  %base.content; >
<!-- end of base.element -->]]>

<!ENTITY % base.attlist  "INCLUDE" >
<![%base.attlist;[
<!ATTLIST %base.qname;
     %XHTML.xmlns.attrib;
     href         %URI.datatype;          #REQUIRED
>
<!-- end of base.attlist -->]]>

<!ENTITY % head.content
   "( %HeadOpts.mix;,
    ( ( %title.qname;, %HeadOpts.mix;, ( %base.qname;, %HeadOpts.mix; )? )
    | ( %base.qname;, %HeadOpts.mix;, ( %title.qname;, %HeadOpts.mix; ))))"
>

<!-- end of xhtml-base-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-base-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Inline Phrasal Module  ..... -->
<!-- file: xhtml-inlphras-1.mod
```

```
   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-inlphras-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Inline Phrasal 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-inlphras-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Inline Phrasal

      abbr, acronym, cite, code, dfn, em, kbd, q, samp, strong, var

   This module declares the elements and their attributes used to
   support inline-level phrasal markup.
-->

<!ENTITY % abbr.element  "INCLUDE" >
<![%abbr.element;[
<!ENTITY % abbr.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % abbr.qname  "abbr" >
<!ELEMENT %abbr.qname;  %abbr.content; >
<!-- end of abbr.element -->]]>

<!ENTITY % abbr.attlist  "INCLUDE" >
<![%abbr.attlist;[
<!ATTLIST %abbr.qname;
    %Common.attrib;
>
<!-- end of abbr.attlist -->]]>

<!ENTITY % acronym.element  "INCLUDE" >
<![%acronym.element;[
<!ENTITY % acronym.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % acronym.qname  "acronym" >
<!ELEMENT %acronym.qname;  %acronym.content; >
<!-- end of acronym.element -->]]>

<!ENTITY % acronym.attlist  "INCLUDE" >
<![%acronym.attlist;[
```

```
<!ATTLIST %acronym.qname;
    %Common.attrib;
>
<!-- end of acronym.attlist -->]]>

<!ENTITY % cite.element  "INCLUDE" >
<![%cite.element;[
<!ENTITY % cite.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % cite.qname  "cite" >
<!ELEMENT %cite.qname;  %cite.content; >
<!-- end of cite.element -->]]>

<!ENTITY % cite.attlist  "INCLUDE" >
<![%cite.attlist;[
<!ATTLIST %cite.qname;
    %Common.attrib;
>
<!-- end of cite.attlist -->]]>

<!ENTITY % code.element  "INCLUDE" >
<![%code.element;[
<!ENTITY % code.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % code.qname  "code" >
<!ELEMENT %code.qname;  %code.content; >
<!-- end of code.element -->]]>

<!ENTITY % code.attlist  "INCLUDE" >
<![%code.attlist;[
<!ATTLIST %code.qname;
    %Common.attrib;
>
<!-- end of code.attlist -->]]>

<!ENTITY % dfn.element  "INCLUDE" >
<![%dfn.element;[
<!ENTITY % dfn.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % dfn.qname  "dfn" >
<!ELEMENT %dfn.qname;  %dfn.content; >
<!-- end of dfn.element -->]]>

<!ENTITY % dfn.attlist  "INCLUDE" >
<![%dfn.attlist;[
```

```
<!ATTLIST %dfn.qname;
    %Common.attrib;
>
<!-- end of dfn.attlist -->]]>

<!ENTITY % em.element  "INCLUDE" >
<![%em.element;[
<!ENTITY % em.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % em.qname  "em" >
<!ELEMENT %em.qname;  %em.content; >
<!-- end of em.element -->]]>

<!ENTITY % em.attlist  "INCLUDE" >
<![%em.attlist;[
<!ATTLIST %em.qname;
    %Common.attrib;
>
<!-- end of em.attlist -->]]>

<!ENTITY % kbd.element  "INCLUDE" >
<![%kbd.element;[
<!ENTITY % kbd.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % kbd.qname  "kbd" >
<!ELEMENT %kbd.qname;  %kbd.content; >
<!-- end of kbd.element -->]]>

<!ENTITY % kbd.attlist  "INCLUDE" >
<![%kbd.attlist;[
<!ATTLIST %kbd.qname;
    %Common.attrib;
>
<!-- end of kbd.attlist -->]]>

<!ENTITY % q.element  "INCLUDE" >
<![%q.element;[
<!ENTITY % q.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % q.qname  "q" >
<!ELEMENT %q.qname;  %q.content; >
<!-- end of q.element -->]]>

<!ENTITY % q.attlist  "INCLUDE" >
<![%q.attlist;[
```

```
<!ATTLIST %q.qname;
    %Common.attrib;
    cite        %URI.datatype;          #IMPLIED
>
<!-- end of q.attlist -->]]>

<!ENTITY % samp.element  "INCLUDE" >
<![%samp.element;[
<!ENTITY % samp.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % samp.qname  "samp" >
<!ELEMENT %samp.qname;  %samp.content; >
<!-- end of samp.element -->]]>

<!ENTITY % samp.attlist  "INCLUDE" >
<![%samp.attlist;[
<!ATTLIST %samp.qname;
    %Common.attrib;
>
<!-- end of samp.attlist -->]]>

<!ENTITY % strong.element  "INCLUDE" >
<![%strong.element;[
<!ENTITY % strong.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % strong.qname  "strong" >
<!ELEMENT %strong.qname;  %strong.content; >
<!-- end of strong.element -->]]>

<!ENTITY % strong.attlist  "INCLUDE" >
<![%strong.attlist;[
<!ATTLIST %strong.qname;
    %Common.attrib;
>
<!-- end of strong.attlist -->]]>

<!ENTITY % var.element  "INCLUDE" >
<![%var.element;[
<!ENTITY % var.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % var.qname  "var" >
<!ELEMENT %var.qname;  %var.content; >
<!-- end of var.element -->]]>

<!ENTITY % var.attlist  "INCLUDE" >
```

<![%var.attlist;[
<!ATTLIST %var.qname;
    %Common.attrib;
>
<!-- end of var.attlist -->]]>

<!-- end of xhtml-inlphras-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-inlphras-1.mod
No license file was found, but licenses were detected in source scan.

SVG 1.1 Tiny DTD

   This is SVG Tiny, a proper subset of SVG.

   The Scalable Vector Graphics (SVG)
   Copyright 2001, 2002 World Wide Web Consortium
     (Massachusetts Institute of Technology, Institut National de
      Recherche en Informatique et en Automatique, Keio University).
      All Rights Reserved.

   Permission to use, copy, modify and distribute the SVG DTD and its
   accompanying documentation for any purpose and without fee is hereby
   granted in perpetuity, provided that the above copyright notice and
   this paragraph appear in all copies. The copyright holders make no
   representation about the suitability of the DTD for any purpose.

   It is provided "as is" without expressed or implied warranty.

     Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
     Revision: $Id: svg11

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg11-tiny.dtd
No license file was found, but licenses were detected in source scan.

<!-- .... -->
<!-- XHTML Basic 1.0 Document Model Module .... -->
<!-- file: xhtml-basic10-model-1.mod

   This is XHTML Basic, a proper subset of XHTML.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-basic10-model-1.mod,v 2.8 2000/11/03 14:28:25 mimasa Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

```
     PUBLIC "-//W3C//ENTITIES XHTML Basic 1.0 Document Model 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-basic10-model-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- XHTML Basic Document Model

   This module describes the groupings of elements that make up
   common content models for XHTML elements.
-->

<!-- Optional Elements in head  ..... -->

<!ENTITY % HeadOpts.mix
   "( %meta.qname; | %link.qname; | %object.qname; )*" >

<!-- Miscellaneous Elements  ..... -->

<!ENTITY % Misc.class "" >

<!-- Inline Elements  ..... -->

<!ENTITY % InlStruct.class "%br.qname; | %span.qname;" >

<!ENTITY % InlPhras.class
   "| %em.qname; | %strong.qname; | %dfn.qname; | %code.qname;
    | %samp.qname; | %kbd.qname; | %var.qname; | %cite.qname;
    | %abbr.qname; | %acronym.qname; | %q.qname;" >

<!ENTITY % InlPres.class "" >

<!ENTITY % I18n.class "" >

<!ENTITY % Anchor.class "| %a.qname;" >

<!ENTITY % InlSpecial.class "| %img.qname; | %object.qname;" >

<!ENTITY % InlForm.class
   "| %input.qname; | %select.qname; | %textarea.qname;
    | %label.qname;"
>

<!ENTITY % Inline.extra "" >

<!ENTITY % Inline.class
```

```
   "%InlStruct.class;
    %InlPhras.class;
    %Anchor.class;
    %InlSpecial.class;
    %InlForm.class;
    %Inline.extra;"
>

<!ENTITY % InlNoAnchor.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlSpecial.class;
    %InlForm.class;
    %Inline.extra;"
>

<!ENTITY % InlNoAnchor.mix
   "%InlNoAnchor.class;
    %Misc.class;"
>

<!ENTITY % Inline.mix
   "%Inline.class;
    %Misc.class;"
>

<!-- Block Elements  ..... -->

<!ENTITY % Heading.class
   "%h1.qname; | %h2.qname; | %h3.qname;
    | %h4.qname; | %h5.qname; | %h6.qname;"
>
<!ENTITY % List.class  "%ul.qname; | %ol.qname; | %dl.qname;" >

<!ENTITY % Table.class "| %table.qname;" >

<!ENTITY % Form.class  "| %form.qname;" >

<!ENTITY % BlkStruct.class "%p.qname; | %div.qname;" >

<!ENTITY % BlkPhras.class
   "| %pre.qname; | %blockquote.qname; | %address.qname;"
>

<!ENTITY % BlkPres.class "" >

<!ENTITY % BlkSpecial.class
   "%Table.class;
```

```
    %Form.class;"
>

<!ENTITY % Block.extra "" >

<!ENTITY % Block.class
   "%BlkStruct.class;
    %BlkPhras.class;
    %BlkSpecial.class;
    %Block.extra;"
>

<!ENTITY % Block.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    %Misc.class;"
>

<!-- All Content Elements  ..... -->

<!-- declares all content except tables
-->
<!ENTITY % FlowNoTable.mix
   "%Heading.class;
    | %List.class;
    | %BlkStruct.class;
    %BlkPhras.class;
    %Form.class;
    %Block.extra;
    | %Inline.class;
    %Misc.class;"
>

<!ENTITY % Flow.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    | %Inline.class;
    %Misc.class;"
>

<!-- end of xhtml-basic10-model-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-basic10-model-1.mod
```

No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Document Model Module ....................................... -->
<!-- file: svg11-model.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg11-model.mod,v 1.3 2002/11/14 15:11:03 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Document Model//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11-model.mod"

   ..................................................................... -->

<!-- SVG 1.1 Document Model

   This module describes the groupings of elements that make up
   common content models for SVG elements.
-->

<!-- module: svg-structure.mod ........................ -->

<!ENTITY % SVG.Description.extra.class "" >
<!ENTITY % SVG.Description.class
  "%SVG.desc.qname; | %SVG.title.qname; | %SVG.metadata.qname;
   %SVG.Description.extra.class;"
>

<!ENTITY % SVG.Use.extra.class "" >
<!ENTITY % SVG.Use.class
  "| %SVG.use.qname; %SVG.Use.extra.class;"
>

<!ENTITY % SVG.Structure.extra.class "" >
<!ENTITY % SVG.Structure.class
  "| %SVG.svg.qname; | %SVG.g.qname; | %SVG.defs.qname; | %SVG.symbol.qname;
     %SVG.Use.class; %SVG.Structure.extra.class;"
>

<!-- module: svg-conditional.mod ....................... -->

<!ENTITY % SVG.Conditional.extra.class "" >
<!ENTITY % SVG.Conditional.class
  "| %SVG.switch.qname; %SVG.Conditional.extra.class;"
>
```

```
<!-- module: svg-image.mod ..... -->

<!ENTITY % SVG.Image.extra.class "" >
<!ENTITY % SVG.Image.class
  "| %SVG.image.qname; %SVG.Image.extra.class;"
>

<!-- module: svg-style.mod ..... -->

<!ENTITY % SVG.Style.extra.class "" >
<!ENTITY % SVG.Style.class
  "| %SVG.style.qname; %SVG.Style.extra.class;"
>

<!-- module: svg-shape.mod ..... -->

<!ENTITY % SVG.Shape.extra.class "" >
<!ENTITY % SVG.Shape.class
  "| %SVG.path.qname; | %SVG.rect.qname; | %SVG.circle.qname;
   | %SVG.line.qname; | %SVG.ellipse.qname; | %SVG.polyline.qname;
   | %SVG.polygon.qname; %SVG.Shape.extra.class;"
>

<!-- module: svg-text.mod ..... -->

<!ENTITY % SVG.Text.extra.class "" >
<!ENTITY % SVG.Text.class
  "| %SVG.text.qname; | %SVG.altGlyphDef.qname; %SVG.Text.extra.class;"
>

<!ENTITY % SVG.TextContent.extra.class "" >
<!ENTITY % SVG.TextContent.class
  "| %SVG.tspan.qname; | %SVG.tref.qname; | %SVG.textPath.qname;
   | %SVG.altGlyph.qname; %SVG.TextContent.extra.class;"
>

<!-- module: svg-marker.mod ..... -->

<!ENTITY % SVG.Marker.extra.class "" >
<!ENTITY % SVG.Marker.class
  "| %SVG.marker.qname; %SVG.Marker.extra.class;"
>

<!-- module: svg-profile.mod ..... -->

<!ENTITY % SVG.ColorProfile.extra.class "" >
<!ENTITY % SVG.ColorProfile.class
```

```
  "| %SVG.color-profile.qname; %SVG.ColorProfile.extra.class;"
>

<!-- module: svg-gradient.mod ..... -->

<!ENTITY % SVG.Gradient.extra.class "" >
<!ENTITY % SVG.Gradient.class
  "| %SVG.linearGradient.qname; | %SVG.radialGradient.qname;
    %SVG.Gradient.extra.class;"
>

<!-- module: svg-pattern.mod ..... -->

<!ENTITY % SVG.Pattern.extra.class "" >
<!ENTITY % SVG.Pattern.class
  "| %SVG.pattern.qname; %SVG.Pattern.extra.class;"
>

<!-- module: svg-clip.mod ..... -->

<!ENTITY % SVG.Clip.extra.class "" >
<!ENTITY % SVG.Clip.class
  "| %SVG.clipPath.qname; %SVG.Clip.extra.class;"
>

<!-- module: svg-mask.mod ..... -->

<!ENTITY % SVG.Mask.extra.class "" >
<!ENTITY % SVG.Mask.class
  "| %SVG.mask.qname; %SVG.Mask.extra.class;"
>

<!-- module: svg-filter.mod ..... -->

<!ENTITY % SVG.Filter.extra.class "" >
<!ENTITY % SVG.Filter.class
  "| %SVG.filter.qname; %SVG.Filter.extra.class;"
>

<!ENTITY % SVG.FilterPrimitive.extra.class "" >
<!ENTITY % SVG.FilterPrimitive.class
  "| %SVG.feBlend.qname; | %SVG.feColorMatrix.qname;
   | %SVG.feComponentTransfer.qname; | %SVG.feComposite.qname;
   | %SVG.feConvolveMatrix.qname; | %SVG.feDiffuseLighting.qname;
   | %SVG.feDisplacementMap.qname; | %SVG.feFlood.qname;
   | %SVG.feGaussianBlur.qname; | %SVG.feImage.qname; | %SVG.feMerge.qname;
   | %SVG.feMorphology.qname; | %SVG.feOffset.qname;
   | %SVG.feSpecularLighting.qname; | %SVG.feTile.qname;
```

```
    | %SVG.feTurbulence.qname; %SVG.FilterPrimitive.extra.class;"
>

<!-- module: svg-cursor.mod ..... -->

<!ENTITY % SVG.Cursor.extra.class "" >
<!ENTITY % SVG.Cursor.class
   "| %SVG.cursor.qname; %SVG.Cursor.extra.class;"
>

<!-- module: svg-hyperlink.mod ..... -->

<!ENTITY % SVG.Hyperlink.extra.class "" >
<!ENTITY % SVG.Hyperlink.class
   "| %SVG.a.qname; %SVG.Hyperlink.extra.class;"
>

<!-- module: svg-view.mod ..... -->

<!ENTITY % SVG.View.extra.class "" >
<!ENTITY % SVG.View.class
   "| %SVG.view.qname; %SVG.View.extra.class;"
>

<!-- module: svg-script.mod ..... -->

<!ENTITY % SVG.Script.extra.class "" >
<!ENTITY % SVG.Script.class
   "| %SVG.script.qname; %SVG.Script.extra.class;"
>

<!-- module: svg-animation.mod ..... -->

<!ENTITY % SVG.Animation.extra.class "" >
<!ENTITY % SVG.Animation.class
   "%SVG.animate.qname; | %SVG.set.qname; | %SVG.animateMotion.qname; |
    %SVG.animateColor.qname; | %SVG.animateTransform.qname;
    %SVG.Animation.extra.class;"
>

<!-- module: svg-font.mod ..... -->

<!ENTITY % SVG.Font.extra.class "" >
<!ENTITY % SVG.Font.class
   "| %SVG.font.qname; | %SVG.font-face.qname; %SVG.Font.extra.class;"
>

<!-- module: svg-extensibility.mod ..... -->
```

```
<!ENTITY % SVG.Extensibility.extra.class "" >
<!ENTITY % SVG.Extensibility.class
  "| %SVG.foreignObject.qname; %SVG.Extensibility.extra.class;"
>

<!-- end of svg11-model.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg11-model.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Param Element Module  ..................................... -->
<!-- file: xhtml-param-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-param-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Param Element 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-param-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Parameters for Java Applets and Embedded Objects

      param

   This module provides declarations for the param element,
   used to provide named property values for the applet
   and object elements.
-->

<!-- param: Named Property Value ..... -->

<!ENTITY % param.element  "INCLUDE" >
<![%param.element;[
<!ENTITY % param.content  "EMPTY" >
<!ENTITY % param.qname  "param" >
<!ELEMENT %param.qname;  %param.content; >
<!-- end of param.element -->]]>
```

```
<!ENTITY % param.attlist  "INCLUDE" >
<![%param.attlist;[
<!ATTLIST %param.qname;
    %XHTML.xmlns.attrib;
    %id.attrib;
    name        CDATA                #REQUIRED
    value       CDATA                #IMPLIED
    valuetype    ( data | ref | object )  'data'
    type        %ContentType.datatype;   #IMPLIED
>
<!-- end of param.attlist -->]]>

<!-- end of xhtml-param-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-param-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Document Style Sheet Module  ..................................... -->
<!-- file: xhtml-style-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-style-1.mod,v 4.1 2001/04/05 06:57:40 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//DTD XHTML Style Sheets 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-style-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Style Sheets

      style

   This module declares the style element type and its attributes,
   used to embed style sheet information in the document head element.
-->

<!-- style: Style Sheet Information ..... -->

<!ENTITY % style.element  "INCLUDE" >
<![%style.element;[
```

```
<!ENTITY % style.content  "( #PCDATA )" >
<!ENTITY % style.qname  "style" >
<!ELEMENT %style.qname;  %style.content; >
<!-- end of style.element -->]]>

<!ENTITY % style.attlist  "INCLUDE" >
<![%style.attlist;[
<!ATTLIST %style.qname;
    %XHTML.xmlns.attrib;
    %id.attrib;
    %title.attrib;
    %I18n.attrib;
    xml:space    ( preserve )            #FIXED 'preserve'
    type        %ContentType.datatype;   #REQUIRED
    media       %MediaDesc.datatype;     #IMPLIED
>
<!-- end of style.attlist -->]]>

<!-- end of xhtml-style-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-style-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Text Module  ..................................................... -->
<!-- file: xhtml-text-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-text-1.mod,v 1.1 2001/02/13 12:24:23 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Text 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-text-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Textual Content

   The Text module includes declarations for all core
   text container elements and their attributes.
-->
```

```
<!ENTITY % xhtml-inlstruct.module "INCLUDE" >
<![%xhtml-inlstruct.module;[
<!ENTITY % xhtml-inlstruct.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Inline Structural 1.0//EN"
         "xhtml-inlstruct-1.mod" >
%xhtml-inlstruct.mod;]]>

<!ENTITY % xhtml-inlphras.module "INCLUDE" >
<![%xhtml-inlphras.module;[
<!ENTITY % xhtml-inlphras.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Inline Phrasal 1.0//EN"
         "xhtml-inlphras-1.mod" >
%xhtml-inlphras.mod;]]>

<!ENTITY % xhtml-blkstruct.module "INCLUDE" >
<![%xhtml-blkstruct.module;[
<!ENTITY % xhtml-blkstruct.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Block Structural 1.0//EN"
         "xhtml-blkstruct-1.mod" >
%xhtml-blkstruct.mod;]]>

<!ENTITY % xhtml-blkphras.module "INCLUDE" >
<![%xhtml-blkphras.module;[
<!ENTITY % xhtml-blkphras.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Block Phrasal 1.0//EN"
         "xhtml-blkphras-1.mod" >
%xhtml-blkphras.mod;]]>

<!-- end of xhtml-text-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-text-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Basic Graphics Attribute Module ............................. -->
<!-- file: svg-basic-graphics-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-basic-graphics-attrib.mod,v 1.2 2002/04/20 18:07:42 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Basic Graphics Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-graphics-attrib.mod"
```

..... -->

<!-- Basic Graphics Attribute

display, visibility

This module defines the Graphics attribute set.
-->

<!ENTITY % SVG.display.attrib
"display (inline | block | list-item | run-in | compact | marker |
table | inline-table | table-row-group | table-header-group |
table-footer-group | table-row | table-column-group |
table-column | table-cell | table-caption | none | inherit)
#IMPLIED"
>

<!ENTITY % SVG.visibility.attrib
"visibility (visible | hidden | inherit) #IMPLIED"
>

<!ENTITY % SVG.Graphics.extra.attrib "" >

<!ENTITY % SVG.Graphics.attrib
"%SVG.display.attrib;
%SVG.visibility.attrib;
%SVG.Graphics.extra.attrib;"
>

<!-- end of svg-basic-graphics-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-basic-graphics-attrib.mod
No license file was found, but licenses were detected in source scan.

SVG 1.1 Basic DTD

This is SVG Basic, a proper subset of SVG.

The Scalable Vector Graphics (SVG)
Copyright 2001, 2002 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the SVG DTD and its
accompanying documentation for any purpose and without fee is hereby

granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg11-basic.dtd
No license file was found, but licenses were detected in source scan.

<!-- .... -->
<!-- XHTML Base Element Module ..... -->
<!-- file: xhtml-base-1.mod

This is XHTML, a reformulation of HTML as a modular XML application.
Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: xhtml-base-1.mod,v 1.1 2001/02/13 12:24:21 ht Exp $ SMI

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS XHTML Base Element 1.0//EN"
SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-base-1.mod"

Revisions:
(none)
..... -->

<!-- Base element

base

This module declares the base element type and its attributes,
used to define a base URI against which relative URIs in the
document will be resolved.

Note that this module also redeclares the content model for
the head element to include the base element.
-->

<!-- base: Document Base URI ..... -->

<!ENTITY % base.element "INCLUDE" >
<![%base.element;[
<!ENTITY % base.content "EMPTY" >

```
<!ENTITY % base.qname  "base" >
<!ELEMENT %base.qname;  %base.content; >
<!-- end of base.element -->]]>

<!ENTITY % base.attlist  "INCLUDE" >
<![%base.attlist;[
<!ATTLIST %base.qname;
    %XHTML.xmlns.attrib;
    href        %URI.datatype;          #REQUIRED
>
<!-- end of base.attlist -->]]>

<!ENTITY % head.content
  "( %HeadOpts.mix;,
   ( ( %title.qname;, %HeadOpts.mix;, ( %base.qname;, %HeadOpts.mix; )? )
   | ( %base.qname;, %HeadOpts.mix;, ( %title.qname;, %HeadOpts.mix; ))))"
>

<!-- end of xhtml-base-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-base-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Block Phrasal Module  ..... -->
<!-- file: xhtml-blkphras-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-blkphras-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Block Phrasal 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-blkphras-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Block Phrasal

      address, blockquote, pre, h1, h2, h3, h4, h5, h6

   This module declares the elements and their attributes used to
   support block-level phrasal markup.
```

```
-->

<!ENTITY % address.element  "INCLUDE" >
<![%address.element;[
<!ENTITY % address.content
   "( #PCDATA | %Inline.mix; )*" >
<!ENTITY % address.qname  "address" >
<!ELEMENT %address.qname;  %address.content; >
<!-- end of address.element -->]]>

<!ENTITY % address.attlist  "INCLUDE" >
<![%address.attlist;[
<!ATTLIST %address.qname;
    %Common.attrib;
>
<!-- end of address.attlist -->]]>

<!ENTITY % blockquote.element  "INCLUDE" >
<![%blockquote.element;[
<!ENTITY % blockquote.content
   "( %Block.mix; )*"
>
<!ENTITY % blockquote.qname  "blockquote" >
<!ELEMENT %blockquote.qname;  %blockquote.content; >
<!-- end of blockquote.element -->]]>

<!ENTITY % blockquote.attlist  "INCLUDE" >
<![%blockquote.attlist;[
<!ATTLIST %blockquote.qname;
    %Common.attrib;
    cite        %URI.datatype;          #IMPLIED
>
<!-- end of blockquote.attlist -->]]>

<!ENTITY % pre.element  "INCLUDE" >
<![%pre.element;[
<!ENTITY % pre.content
   "( #PCDATA
    | %InlStruct.class;
    %InlPhras.class;
    | %tt.qname; | %i.qname; | %b.qname;
    %I18n.class;
    %Anchor.class;
    | %map.qname;
    %Misc.class;
    %Inline.extra; )*"
>
<!ENTITY % pre.qname  "pre" >
```

```
<!ELEMENT %pre.qname;  %pre.content; >
<!-- end of pre.element -->]]>

<!ENTITY % pre.attlist  "INCLUDE" >
<![%pre.attlist;[
<!ATTLIST %pre.qname;
    %Common.attrib;
>
<!-- end of pre.attlist -->]]>

<!-- .....  Heading Elements  ..... -->

<!ENTITY % Heading.content  "( #PCDATA | %Inline.mix; )*" >

<!ENTITY % h1.element  "INCLUDE" >
<![%h1.element;[
<!ENTITY % h1.qname  "h1" >
<!ELEMENT %h1.qname;  %Heading.content; >
<!-- end of h1.element -->]]>

<!ENTITY % h1.attlist  "INCLUDE" >
<![%h1.attlist;[
<!ATTLIST %h1.qname;
    %Common.attrib;
>
<!-- end of h1.attlist -->]]>

<!ENTITY % h2.element  "INCLUDE" >
<![%h2.element;[
<!ENTITY % h2.qname  "h2" >
<!ELEMENT %h2.qname;  %Heading.content; >
<!-- end of h2.element -->]]>

<!ENTITY % h2.attlist  "INCLUDE" >
<![%h2.attlist;[
<!ATTLIST %h2.qname;
    %Common.attrib;
>
<!-- end of h2.attlist -->]]>

<!ENTITY % h3.element  "INCLUDE" >
<![%h3.element;[
<!ENTITY % h3.qname  "h3" >
<!ELEMENT %h3.qname;  %Heading.content; >
<!-- end of h3.element -->]]>

<!ENTITY % h3.attlist  "INCLUDE" >
<![%h3.attlist;[
```

```
<!ATTLIST %h3.qname;
    %Common.attrib;
>
<!-- end of h3.attlist -->]]>

<!ENTITY % h4.element  "INCLUDE" >
<![%h4.element;[
<!ENTITY % h4.qname  "h4" >
<!ELEMENT %h4.qname;  %Heading.content; >
<!-- end of h4.element -->]]>

<!ENTITY % h4.attlist  "INCLUDE" >
<![%h4.attlist;[
<!ATTLIST %h4.qname;
    %Common.attrib;
>
<!-- end of h4.attlist -->]]>

<!ENTITY % h5.element  "INCLUDE" >
<![%h5.element;[
<!ENTITY % h5.qname  "h5" >
<!ELEMENT %h5.qname;  %Heading.content; >
<!-- end of h5.element -->]]>

<!ENTITY % h5.attlist  "INCLUDE" >
<![%h5.attlist;[
<!ATTLIST %h5.qname;
    %Common.attrib;
>
<!-- end of h5.attlist -->]]>

<!ENTITY % h6.element  "INCLUDE" >
<![%h6.element;[
<!ENTITY % h6.qname  "h6" >
<!ELEMENT %h6.qname;  %Heading.content; >
<!-- end of h6.element -->]]>

<!ENTITY % h6.attlist  "INCLUDE" >
<![%h6.attlist;[
<!ATTLIST %h6.qname;
    %Common.attrib;
>
<!-- end of h6.attlist -->]]>

<!-- end of xhtml-blkphras-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
```

jar/net/sf/saxon/data/w3c/xhtml11/xhtml-blkphras-1.mod
No license file was found, but licenses were detected in source scan.

2005 World Wide Web Consortium
(Massachusetts Institute of Technology, European Research Consortium
for Informatics and Mathematics, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the XHTML Basic DTD
and its accompanying documentation for any purpose and without fee is
hereby granted in perpetuity, provided that the above copyright notice
and this paragraph appear in all copies. The copyright holders make
no representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Editors: Murray M. Altheim <mailto:altheim@eng.sun.com>
Peter Stark <mailto:Peter.Stark@ecs.ericsson.se>
Revision: $Id: xhtml

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml10/xhtml-basic10.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Access Module ..... -->
<!-- file: xhtml-access-1.mod

This is XHTML Access - the Access Module for XHTML.

Copyright 2007-2008 W3C (MIT, ERCIM, Keio), All Rights Reserved.

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS XHTML Access Element 1.0//EN"
SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-access-1.mod"

Revisions:
(none)
..... -->

<!ENTITY % Character.datatype "CDATA" >
<!ENTITY % CURIEs.datatype "CDATA" >
<!ENTITY % IDREFs.datatype "CDATA" >

<!ENTITY % access.element "INCLUDE" >

```
<![%access.element;[
<!ENTITY % access.content  "EMPTY" >
<!ENTITY % XHTML-ACCESS.access.qname  "access" >
<!ELEMENT %XHTML-ACCESS.access.qname;  %access.content; >
<!-- end of access.element -->]]>

<!ENTITY % access.attlist  "INCLUDE" >
<![%access.attlist;[
<!ATTLIST %access.qname;
    %Common.attrib;
    activate     ( yes | no )            #IMPLIED
    key          %Character.datatype;     #IMPLIED
    targetid     %IDREFs.datatype;        #IMPLIED
    targetrole   %CURIEs.datatype;        #IMPLIED
>
<!-- end of access.attlist -->]]>

<!-- end of xhtml-access-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-access-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Marker Module ..................................................... -->
<!-- file: svg-marker.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-marker.mod,v 1.4 2002/11/14 15:11:03 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Marker//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-marker.mod"

   ..................................................................... -->

<!-- Marker

     marker

   This module declares markup to provide support for marker.
-->

<!-- 'marker' property/attribute value (e.g., 'none', <uri>) -->
<!ENTITY % MarkerValue.datatype "CDATA" >
```

```
<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.marker.qname "marker" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Marker.class ..... -->

<!ENTITY % SVG.Marker.extra.class "" >

<!ENTITY % SVG.Marker.class
  "| %SVG.marker.qname; %SVG.Marker.extra.class;"
>

<!-- SVG.Marker.attrib ..... -->

<!ENTITY % SVG.Marker.extra.attrib "" >

<!ENTITY % SVG.Marker.attrib
  "marker-start %MarkerValue.datatype; #IMPLIED
   marker-mid %MarkerValue.datatype; #IMPLIED
   marker-end %MarkerValue.datatype; #IMPLIED
   %SVG.Marker.extra.attrib;"
>

<!-- SVG.Presentation.attrib ..... -->
```

```
<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
   %SVG.Viewport.attrib;
   %SVG.Text.attrib;
   %SVG.TextContent.attrib;
   %SVG.Font.attrib;
   %SVG.Paint.attrib;
   %SVG.Color.attrib;
   %SVG.Opacity.attrib;
   %SVG.Graphics.attrib;
   %SVG.Marker.attrib;
   %SVG.ColorProfile.attrib;
   %SVG.Gradient.attrib;
   %SVG.Clip.attrib;
   %SVG.Mask.attrib;
   %SVG.Filter.attrib;
   %SVG.FilterColor.attrib;
   %SVG.Cursor.attrib;
   flood-color %SVGColor.datatype; #IMPLIED
   flood-opacity %OpacityValue.datatype; #IMPLIED
   lighting-color %SVGColor.datatype; #IMPLIED
   %SVG.Presentation.extra.attrib;"
>

<!-- marker: Marker Element ..... -->

<!ENTITY % SVG.marker.extra.content "" >

<!ENTITY % SVG.marker.element "INCLUDE" >
<![%SVG.marker.element;[
<!ENTITY % SVG.marker.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
     %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
     %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
     %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
     %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
     %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
     %SVG.Font.class; %SVG.marker.extra.content; )*"
>
<!ELEMENT %SVG.marker.qname; %SVG.marker.content; >
<!-- end of SVG.marker.element -->]]>

<!ENTITY % SVG.marker.attlist "INCLUDE" >
<![%SVG.marker.attlist;[
<!ATTLIST %SVG.marker.qname;
  %SVG.Core.attrib;
```

```
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.External.attrib;
  refX %Coordinate.datatype; #IMPLIED
  refY %Coordinate.datatype; #IMPLIED
  markerUnits ( strokeWidth | userSpaceOnUse ) #IMPLIED
  markerWidth  %Length.datatype; #IMPLIED
  markerHeight %Length.datatype; #IMPLIED
  orient CDATA #IMPLIED
  viewBox %ViewBoxSpec.datatype; #IMPLIED
  preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
>
<!-- end of SVG.marker.attlist -->]]>

<!-- end of svg-marker.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-marker.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Gradient Module ..................................... -->
<!-- file: svg-gradient.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-gradient.mod,v 1.3 2002/10/24 17:40:15 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Gradient//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-gradient.mod"

   ..................................................................... -->

<!-- Gradient

     linearGradient, radialGradient, stop

   This module declares markup to provide support for gradient fill.
-->

<!-- a <number> or a <percentage> -->
<!ENTITY % NumberOrPercentage.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->
```

```
<!ENTITY % SVG.linearGradient.qname "linearGradient" >
<!ENTITY % SVG.radialGradient.qname "radialGradient" >
<!ENTITY % SVG.stop.qname "stop" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Gradient.class ..... -->

<!ENTITY % SVG.Gradient.extra.class "" >

<!ENTITY % SVG.Gradient.class
  "| %SVG.linearGradient.qname; | %SVG.radialGradient.qname;
    %SVG.Gradient.extra.class;"
>

<!-- SVG.Gradient.attrib ..... -->

<!ENTITY % SVG.Gradient.extra.attrib "" >

<!ENTITY % SVG.Gradient.attrib
  "stop-color %SVGColor.datatype; #IMPLIED
   stop-opacity %OpacityValue.datatype; #IMPLIED
   %SVG.Gradient.extra.attrib;"
>

<!-- linearGradient: Linear Gradient Element ..... -->

<!ENTITY % SVG.linearGradient.extra.content "" >

<!ENTITY % SVG.linearGradient.element "INCLUDE" >
<![%SVG.linearGradient.element;[
<!ENTITY % SVG.linearGradient.content
  "(( %SVG.Description.class; )*, ( %SVG.stop.qname; | %SVG.animate.qname;
    | %SVG.set.qname; | %SVG.animateTransform.qname;
      %SVG.linearGradient.extra.content; )*)"
>
<!ELEMENT %SVG.linearGradient.qname; %SVG.linearGradient.content; >
<!-- end of SVG.linearGradient.element -->]]>

<!ENTITY % SVG.linearGradient.attlist "INCLUDE" >
<![%SVG.linearGradient.attlist;[
<!ATTLIST %SVG.linearGradient.qname;
```

```
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Color.attrib;
  %SVG.Gradient.attrib;
  %SVG.XLink.attrib;
  %SVG.External.attrib;
  x1 %Coordinate.datatype; #IMPLIED
  y1 %Coordinate.datatype; #IMPLIED
  x2 %Coordinate.datatype; #IMPLIED
  y2 %Coordinate.datatype; #IMPLIED
  gradientUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
  gradientTransform %TransformList.datatype; #IMPLIED
  spreadMethod ( pad | reflect | repeat ) #IMPLIED
>
<!-- end of SVG.linearGradient.attlist -->]]>

<!-- radialGradient: Radial Gradient Element ..... -->

<!ENTITY % SVG.radialGradient.extra.content "" >

<!ENTITY % SVG.radialGradient.element "INCLUDE" >
<![%SVG.radialGradient.element;[
<!ENTITY % SVG.radialGradient.content
  "(( %SVG.Description.class; )*, ( %SVG.stop.qname; | %SVG.animate.qname;
    | %SVG.set.qname; | %SVG.animateTransform.qname;
      %SVG.radialGradient.extra.content; )*)"
>
<!ELEMENT %SVG.radialGradient.qname; %SVG.radialGradient.content; >
<!-- end of SVG.radialGradient.element -->]]>

<!ENTITY % SVG.radialGradient.attlist "INCLUDE" >
<![%SVG.radialGradient.attlist;[
<!ATTLIST %SVG.radialGradient.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Color.attrib;
  %SVG.Gradient.attrib;
  %SVG.XLink.attrib;
  %SVG.External.attrib;
  cx %Coordinate.datatype; #IMPLIED
  cy %Coordinate.datatype; #IMPLIED
  r %Length.datatype; #IMPLIED
  fx %Coordinate.datatype; #IMPLIED
  fy %Coordinate.datatype; #IMPLIED
  gradientUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
  gradientTransform %TransformList.datatype; #IMPLIED
  spreadMethod ( pad | reflect | repeat ) #IMPLIED
>
```

```
<!-- end of SVG.radialGradient.attlist -->]]>

<!-- stop: Stop Element ..... -->

<!ENTITY % SVG.stop.extra.content "" >

<!ENTITY % SVG.stop.element "INCLUDE" >
<![%SVG.stop.element;[
<!ENTITY % SVG.stop.content
  "( %SVG.animate.qname; | %SVG.set.qname; | %SVG.animateColor.qname;
    %SVG.stop.extra.content; )*"
>
<!ELEMENT %SVG.stop.qname; %SVG.stop.content; >
<!-- end of SVG.stop.element -->]]>

<!ENTITY % SVG.stop.attlist "INCLUDE" >
<![%SVG.stop.attlist;[
<!ATTLIST %SVG.stop.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Color.attrib;
  %SVG.Gradient.attrib;
  offset %NumberOrPercentage.datatype; #REQUIRED
>
<!-- end of SVG.stop.attlist -->]]>

<!-- end of svg-gradient.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-gradient.mod
No license file was found, but licenses were detected in source scan.

dev/

```
    Permission to use, copy, modify and distribute the RDDL DTD
    and its accompanying documentation for any purpose and without fee is
    hereby granted in perpetuity, provided that the above copyright notice
    and this paragraph appear in all copies.  The copyright holders make
    no representation about the suitability of the DTD for any purpose.

    It is provided "as is" without expressed or implied warranty.

  This is the driver file for version 1.0 of the RDDL 1.0 DTD as an extension of XHTML Basic 1.0.

    This DTD is identified by the PUBLIC and SYSTEM identifiers:

    PUBLIC: "
```

2000 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the XHTML Basic DTD
and its accompanying documentation for any purpose and without fee is
hereby granted in perpetuity, provided that the above copyright notice
and this paragraph appear in all copies. The copyright holders make
no representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Editors: Murray M. Altheim <mailto:altheim@eng.sun.com>
Peter Stark <mailto:Peter.Stark@ecs.ericsson.se>
Revision: $Id: rddl

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/rddl-xhtml.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Notations Module ..... -->
<!-- file: xhtml-notations-1.mod

This is XHTML, a reformulation of HTML as a modular XML application.
Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: xhtml-notations-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//NOTATIONS XHTML Notations 1.0//EN"
SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-notations-1.mod"

Revisions:
(none)
..... -->

<!-- Notations

defines the following notations, many of these imported from
other specifications and standards. When an existing FPI is
known, it is incorporated here.
-->

<!-- XML Notations ..... -->
<!-- SGML and XML Notations ..... -->

```
<!-- W3C XML 1.0 Recommendation -->
<!NOTATION w3c-xml
   PUBLIC "ISO 8879//NOTATION Extensible Markup Language (XML) 1.0//EN" >

<!-- XML 1.0 CDATA -->
<!NOTATION cdata
   PUBLIC "-//W3C//NOTATION XML 1.0: CDATA//EN" >

<!-- SGML Formal Public Identifiers -->
<!NOTATION fpi
   PUBLIC "ISO 8879:1986//NOTATION Formal Public Identifier//EN" >

<!-- XHTML Notations ..... -->

<!-- Length defined for cellpadding/cellspacing -->

<!-- nn for pixels or nn% for percentage length -->
<!NOTATION length
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Length//EN" >

<!-- space-separated list of link types -->
<!NOTATION linkTypes
  PUBLIC "-//W3C//NOTATION XHTML Datatype: LinkTypes//EN" >

<!-- single or comma-separated list of media descriptors -->
<!NOTATION mediaDesc
  PUBLIC "-//W3C//NOTATION XHTML Datatype: MediaDesc//EN" >

<!-- pixel, percentage, or relative -->
<!NOTATION multiLength
  PUBLIC "-//W3C//NOTATION XHTML Datatype: MultiLength//EN" >

<!-- one or more digits (NUMBER) -->
<!NOTATION number
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Number//EN" >

<!-- integer representing length in pixels -->
<!NOTATION pixels
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Pixels//EN" >

<!-- script expression -->
<!NOTATION script
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Script//EN" >

<!-- textual content -->
<!NOTATION text
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Text//EN" >
```

```
<!-- Imported Notations ..... -->

<!-- a single character from [ISO10646] -->
<!NOTATION character
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Character//EN" >

<!-- a character encoding, as per [RFC2045] -->
<!NOTATION charset
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Charset//EN" >

<!-- a space separated list of character encodings, as per [RFC2045] -->
<!NOTATION charsets
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Charsets//EN" >

<!-- media type, as per [RFC2045] -->
<!NOTATION contentType
  PUBLIC "-//W3C//NOTATION XHTML Datatype: ContentType//EN" >

<!-- comma-separated list of media types, as per [RFC2045] -->
<!NOTATION contentTypes
  PUBLIC "-//W3C//NOTATION XHTML Datatype: ContentTypes//EN" >

<!-- date and time information. ISO date format -->
<!NOTATION datetime
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Datetime//EN" >

<!-- a language code, as per [RFC1766] -->
<!NOTATION languageCode
  PUBLIC "-//W3C//NOTATION XHTML Datatype: LanguageCode//EN" >

<!-- a Uniform Resource Identifier, see [URI] -->
<!NOTATION uri
  PUBLIC "-//W3C//NOTATION XHTML Datatype: URI//EN" >

<!-- a space-separated list of Uniform Resource Identifiers, see [URI] -->
<!NOTATION uris
  PUBLIC "-//W3C//NOTATION XHTML Datatype: URIs//EN" >

<!-- end of xhtml-notations-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-notations-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..... -->
<!-- XHTML Legacy Redeclarations Module ..... -->
```

```
<!-- file: xhtml-legacy-redecl-1.mod

   This is an extension of XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-legacy-redecl-1.mod,v 4.4 2001/04/10 09:42:30 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Legacy Redeclarations 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-legacy-redecl-1.mod"

   Revisions:
   (none)
   ..... -->


<!-- Legacy Redeclarations

   This optional module replaces the Modular Framework module, interspersing
   redeclarations of various parameter entities to allow for inclusions of
   Transitional markup in the XHTML 1.1 document model. This instantiates
   the modules needed to support the XHTML modularization model, including:

      + notations
      + datatypes
      + namespace-qualified names
      + common attributes
      + document model
      + character entities

   By default, the Object module is included, with Frames and IFrames ignored.

   The Intrinsic Events module is ignored by default but
   occurs in this module because it must be instantiated
   prior to Attributes but after Datatypes.
-->
<!ENTITY % xhtml-legacy.module "INCLUDE" >

<!ENTITY % xhtml-arch.module "IGNORE" >
<![%xhtml-arch.module;[
<!ENTITY % xhtml-arch.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Base Architecture 1.0//EN"
         "xhtml-arch-1.mod" >
%xhtml-arch.mod;]]>

<!ENTITY % xhtml-notations.module "INCLUDE" >
<![%xhtml-notations.module;[
<!ENTITY % xhtml-notations.mod
```

```
   PUBLIC "-//W3C//NOTATIONS XHTML Notations 1.0//EN"
         "xhtml-notations-1.mod" >
%xhtml-notations.mod;]]>

<!-- Datatypes Module ..... -->
<!ENTITY % xhtml-datatypes.module "INCLUDE" >
<![%xhtml-datatypes.module;[
<!ENTITY % xhtml-datatypes.mod
   PUBLIC "-//W3C//ENTITIES XHTML Datatypes 1.0//EN"
         "xhtml-datatypes-1.mod" >
%xhtml-datatypes.mod;]]>

<!-- Qualified Names Module ..... -->
<!ENTITY % xhtml-qname.module "INCLUDE" >
<![%xhtml-qname.module;[
<!ENTITY % xhtml-qname.mod
   PUBLIC "-//W3C//ENTITIES XHTML Qualified Names 1.0//EN"
         "xhtml-qname-1.mod" >
%xhtml-qname.mod;]]>

<!-- Additional Qualified Names ..... -->

<!-- xhtml-legacy-1.mod -->
<!ENTITY % font.qname     "%XHTML.pfx;font" >
<!ENTITY % basefont.qname "%XHTML.pfx;basefont" >
<!ENTITY % center.qname   "%XHTML.pfx;center" >
<!ENTITY % s.qname        "%XHTML.pfx;s" >
<!ENTITY % strike.qname   "%XHTML.pfx;strike" >
<!ENTITY % u.qname        "%XHTML.pfx;u" >
<!ENTITY % dir.qname      "%XHTML.pfx;dir" >
<!ENTITY % menu.qname     "%XHTML.pfx;menu" >
<!ENTITY % isindex.qname  "%XHTML.pfx;isindex" >

<!-- xhtml-frames-1.mod -->
<!ENTITY % frameset.qname "%XHTML.pfx;frameset" >
<!ENTITY % frame.qname    "%XHTML.pfx;frame" >
<!ENTITY % noframes.qname "%XHTML.pfx;noframes" >

<!-- xhtml-iframe-1.mod -->
<!ENTITY % iframe.qname   "%XHTML.pfx;iframe" >

<!ENTITY % xhtml-events.module "IGNORE" >
<![%xhtml-events.module;[
<!ENTITY % xhtml-events.mod
   PUBLIC "-//W3C//ENTITIES XHTML Intrinsic Events 1.0//EN"
         "xhtml-events-1.mod" >
%xhtml-events.mod;]]>
```

```
<!-- Additional Common Attributes ..... -->

<!-- include historical 'lang' attribute (which should
   always match the value of 'xml:lang')
-->
<!ENTITY % lang.attrib
   "xml:lang     %LanguageCode.datatype;  #IMPLIED
    lang         %LanguageCode.datatype;  #IMPLIED"
>

<!-- Common Attributes Module ..... -->
<!ENTITY % xhtml-attribs.module "INCLUDE" >
<![%xhtml-attribs.module;[
<!ENTITY % xhtml-attribs.mod
   PUBLIC "-//W3C//ENTITIES XHTML Common Attributes 1.0//EN"
         "xhtml-attribs-1.mod" >
%xhtml-attribs.mod;]]>

<!-- placeholder for content model redeclarations -->
<!ENTITY % xhtml-model.redecl "" >
%xhtml-model.redecl;

<!-- Document Model Redeclarations ..... -->

<!ENTITY % InlPres.class
   "| %tt.qname; | %i.qname; | %b.qname; | %big.qname;
    | %small.qname; | %sub.qname; | %sup.qname;
    | %font.qname; | %basefont.qname; | %iframe.qname;
    | %s.qname; | %strike.qname; | %u.qname;"
>

<!ENTITY % InlSpecial.class
   "| %img.qname; | %map.qname;
    | %applet.qname; | %object.qname;" >

<!ENTITY % BlkPres.class
   "| %hr.qname; | %center.qname;"
>

<!ENTITY % BlkSpecial.class
   "| %table.qname; | %form.qname; | %fieldset.qname;
    | %noframes.qname; | %isindex.qname;"
>

<!ENTITY % List.class
   "%ul.qname; | %ol.qname; | %dl.qname;
    | %dir.qname; | %menu.qname;"
>
```

```
<!-- Document Model Module  ..... -->
<!ENTITY % xhtml-model.module "INCLUDE" >
<![%xhtml-model.module;[
<!-- instantiate the Document Model module declared in the DTD driver
-->
%xhtml-model.mod;]]>

<!ENTITY % blockquote.content
   "( #PCDATA | %Flow.mix; )*"
>

<!ENTITY % noscript.content
    "( #PCDATA | %Flow.mix; )*"
>

<!ENTITY % body.content
   "( #PCDATA | %Flow.mix; )*"
>

<!-- redeclare content model of <html> to allow for either
   body or frameset content. The SGML markup minimization
   features used in HTML 4 do not apply, so the ambiguity
   that necessitated separation into the separate Frameset
   and Transitional DTDs is eliminated.
-->
<!ENTITY % html.content
   "( %head.qname;, ( %body.qname; | %frameset.qname; ) )"
>



<!ENTITY % xhtml-charent.module "INCLUDE" >
<![%xhtml-charent.module;[
<!ENTITY % xhtml-charent.mod
   PUBLIC "-//W3C//ENTITIES XHTML Character Entities 1.0//EN"
        "xhtml-charent-1.mod" >
%xhtml-charent.mod;]]>

<!-- end of xhtml-legacy-redecl-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-legacy-redecl-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML IFrame Module  ..... -->
<!-- file: xhtml-iframe-1.mod
```

```
   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-iframe-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Inline Frame Element 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-iframe-1.mod"

   Revisions:
#2000-10-22: added #PCDATA to <iframe> content model as per HTML 4
   ..................................................................... -->

<!-- Inline Frames

      iframe

   This module declares the iframe element type and its attributes,
   used to create an inline frame within a document.
-->

<!-- Inline Frames ..................................... -->

<!ENTITY % iframe.content  "( #PCDATA | %Flow.mix; )*" >
<!ENTITY % iframe.qname  "iframe" >
<!ELEMENT %iframe.qname;  %iframe.content; >
<!ATTLIST %iframe.qname;
    %Core.attrib;
    longdesc     %URI.datatype;          #IMPLIED
    src          %URI.datatype;          #IMPLIED
    frameborder  ( 1 | 0 )               '1'
    marginwidth  %Pixels.datatype;       #IMPLIED
    marginheight %Pixels.datatype;       #IMPLIED
    scrolling    ( yes | no | auto )     'auto'
    height       %Length.datatype;       #IMPLIED
    width        %Length.datatype;       #IMPLIED
>

<!-- end of xhtml-iframe-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-iframe-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- MathML Qualified Names Module  ..................................... -->
```

```
<!-- file: mathml3-qname-1.mod

   This is the Mathematical Markup Language (MathML) 2.0, an XML
   application for describing mathematical notation and capturing
   both its structure and content.

   Copyright 1998-2009 W3C (MIT, INRIA, Keio), All Rights Reserved.

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES MathML 3.0 Qualified Names 1.0//EN"
     SYSTEM "mathml3-qname.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- MathML Qualified Names

   This module is contained in two parts, labeled Section 'A' and 'B':

     Section A declares parameter entities to support namespace-
     qualified names, namespace declarations, and name prefixing
     for MathML.

     Section B declares parameter entities used to provide
     namespace-qualified names for all MathML element types.

   This module is derived from the XHTML Qualified Names Template module.
-->

<!-- Section A: XHTML XML Namespace Framework :::::::::::::::::::: -->

<!ENTITY % NS.prefixed     "IGNORE" >
<!ENTITY % MATHML.prefixed "%NS.prefixed;" >

<!-- XLink ............... -->

<!ENTITY % XLINK.prefix        "xlink" >
<!ENTITY % XLINK.xmlns "http://www.w3.org/1999/xlink" >
<!ENTITY % XLINK.xmlns.attrib
   "xmlns:%XLINK.prefix;  CDATA          #FIXED '%XLINK.xmlns;'"
>

<!-- W3C XML Schema ............... -->

<!ENTITY % Schema.prefix        "xsi" >
<!ENTITY % Schema.xmlns "http://www.w3.org/2001/XMLSchema-instance" >
```

```
<!ENTITY % Schema.xmlns.attrib
   "xmlns:%Schema.prefix;  CDATA          #IMPLIED"
>

<!-- MathML ..... -->

<!ENTITY % MATHML.xmlns    "http://www.w3.org/1998/Math/MathML" >
<!ENTITY % MATHML.prefix   "m" >
<![%MATHML.prefixed;[
<!ENTITY % MATHML.xmlns.extra.attrib  "" >
]]>
<!ENTITY % MATHML.xmlns.extra.attrib
   "%XLINK.xmlns.attrib;
    %Schema.xmlns.attrib;" >

<![%MATHML.prefixed;[
<!ENTITY % MATHML.pfx  "%MATHML.prefix;:" >
<!ENTITY % MATHML.xmlns.attrib
   "xmlns:%MATHML.prefix;  CDATA   #FIXED '%MATHML.xmlns;'
    %MATHML.xmlns.extra.attrib;"
>
]]>
<!ENTITY % MATHML.pfx  "" >
<!ENTITY % MATHML.xmlns.attrib
   "xmlns       CDATA          #FIXED '%MATHML.xmlns;'
    %MATHML.xmlns.extra.attrib;"
>

<![%NS.prefixed;[
<!ENTITY % XHTML.xmlns.extra.attrib
   "%MATHML.xmlns.attrib;" >
]]>
<!ENTITY % XHTML.xmlns.extra.attrib
   "%XLINK.xmlns.attrib;
    %Schema.xmlns.attrib;"
>


<!-- ignores subsequent instantiation of this module when
   used as external subset rather than module fragment.
   NOTE: Do not modify this parameter entity, otherwise
   a recursive parsing situation may result.
-->
<!ENTITY % mathml-qname.module "IGNORE" >

<!-- Section B: MathML Qualified Names ::::::::::::::::::::::::::::: -->

<!-- 9. This section declares parameter entities used to provide
```

```
      namespace-qualified names for all MathML element types.
-->

<!ENTITY % abs.qname "%MATHML.pfx;abs" >
<!ENTITY % and.qname "%MATHML.pfx;and" >
<!ENTITY % annotation-xml.qname "%MATHML.pfx;annotation-xml" >
<!ENTITY % annotation.qname "%MATHML.pfx;annotation" >
<!ENTITY % apply.qname "%MATHML.pfx;apply" >
<!ENTITY % approx.qname "%MATHML.pfx;approx" >
<!ENTITY % arccos.qname "%MATHML.pfx;arccos" >
<!ENTITY % arccosh.qname "%MATHML.pfx;arccosh" >
<!ENTITY % arccot.qname "%MATHML.pfx;arccot" >
<!ENTITY % arccoth.qname "%MATHML.pfx;arccoth" >
<!ENTITY % arccsc.qname "%MATHML.pfx;arccsc" >
<!ENTITY % arccsch.qname "%MATHML.pfx;arccsch" >
<!ENTITY % arcsec.qname "%MATHML.pfx;arcsec" >
<!ENTITY % arcsech.qname "%MATHML.pfx;arcsech" >
<!ENTITY % arcsin.qname "%MATHML.pfx;arcsin" >
<!ENTITY % arcsinh.qname "%MATHML.pfx;arcsinh" >
<!ENTITY % arctan.qname "%MATHML.pfx;arctan" >
<!ENTITY % arctanh.qname "%MATHML.pfx;arctanh" >
<!ENTITY % arg.qname "%MATHML.pfx;arg" >
<!ENTITY % bind.qname "%MATHML.pfx;bind" >
<!ENTITY % bvar.qname "%MATHML.pfx;bvar" >
<!ENTITY % card.qname "%MATHML.pfx;card" >
<!ENTITY % cartesianproduct.qname "%MATHML.pfx;cartesianproduct" >
<!ENTITY % cbytes.qname "%MATHML.pfx;cbytes" >
<!ENTITY % ceiling.qname "%MATHML.pfx;ceiling" >
<!ENTITY % cerror.qname "%MATHML.pfx;cerror" >
<!ENTITY % ci.qname "%MATHML.pfx;ci" >
<!ENTITY % cn.qname "%MATHML.pfx;cn" >
<!ENTITY % codomain.qname "%MATHML.pfx;codomain" >
<!ENTITY % complexes.qname "%MATHML.pfx;complexes" >
<!ENTITY % compose.qname "%MATHML.pfx;compose" >
<!ENTITY % condition.qname "%MATHML.pfx;condition" >
<!ENTITY % conjugate.qname "%MATHML.pfx;conjugate" >
<!ENTITY % cos.qname "%MATHML.pfx;cos" >
<!ENTITY % cosh.qname "%MATHML.pfx;cosh" >
<!ENTITY % cot.qname "%MATHML.pfx;cot" >
<!ENTITY % coth.qname "%MATHML.pfx;coth" >
<!ENTITY % cs.qname "%MATHML.pfx;cs" >
<!ENTITY % csc.qname "%MATHML.pfx;csc" >
<!ENTITY % csch.qname "%MATHML.pfx;csch" >
<!ENTITY % csymbol.qname "%MATHML.pfx;csymbol" >
<!ENTITY % curl.qname "%MATHML.pfx;curl" >
<!ENTITY % declare.qname "%MATHML.pfx;declare" >
<!ENTITY % degree.qname "%MATHML.pfx;degree" >
<!ENTITY % determinant.qname "%MATHML.pfx;determinant" >
```

<!ENTITY % diff.qname "%MATHML.pfx;diff" >
<!ENTITY % divergence.qname "%MATHML.pfx;divergence" >
<!ENTITY % divide.qname "%MATHML.pfx;divide" >
<!ENTITY % domain.qname "%MATHML.pfx;domain" >
<!ENTITY % domainofapplication.qname "%MATHML.pfx;domainofapplication" >
<!ENTITY % emptyset.qname "%MATHML.pfx;emptyset" >
<!ENTITY % eq.qname "%MATHML.pfx;eq" >
<!ENTITY % equivalent.qname "%MATHML.pfx;equivalent" >
<!ENTITY % eulergamma.qname "%MATHML.pfx;eulergamma" >
<!ENTITY % exists.qname "%MATHML.pfx;exists" >
<!ENTITY % exp.qname "%MATHML.pfx;exp" >
<!ENTITY % exponentiale.qname "%MATHML.pfx;exponentiale" >
<!ENTITY % factorial.qname "%MATHML.pfx;factorial" >
<!ENTITY % factorof.qname "%MATHML.pfx;factorof" >
<!ENTITY % false.qname "%MATHML.pfx;false" >
<!ENTITY % floor.qname "%MATHML.pfx;floor" >
<!ENTITY % fn.qname "%MATHML.pfx;fn" >
<!ENTITY % forall.qname "%MATHML.pfx;forall" >
<!ENTITY % gcd.qname "%MATHML.pfx;gcd" >
<!ENTITY % geq.qname "%MATHML.pfx;geq" >
<!ENTITY % grad.qname "%MATHML.pfx;grad" >
<!ENTITY % gt.qname "%MATHML.pfx;gt" >
<!ENTITY % ident.qname "%MATHML.pfx;ident" >
<!ENTITY % image.qname "%MATHML.pfx;image" >
<!ENTITY % imaginary.qname "%MATHML.pfx;imaginary" >
<!ENTITY % imaginaryi.qname "%MATHML.pfx;imaginaryi" >
<!ENTITY % implies.qname "%MATHML.pfx;implies" >
<!ENTITY % in.qname "%MATHML.pfx;in" >
<!ENTITY % infinity.qname "%MATHML.pfx;infinity" >
<!ENTITY % int.qname "%MATHML.pfx;int" >
<!ENTITY % integers.qname "%MATHML.pfx;integers" >
<!ENTITY % intersect.qname "%MATHML.pfx;intersect" >
<!ENTITY % interval.qname "%MATHML.pfx;interval" >
<!ENTITY % inverse.qname "%MATHML.pfx;inverse" >
<!ENTITY % lambda.qname "%MATHML.pfx;lambda" >
<!ENTITY % laplacian.qname "%MATHML.pfx;laplacian" >
<!ENTITY % lcm.qname "%MATHML.pfx;lcm" >
<!ENTITY % leq.qname "%MATHML.pfx;leq" >
<!ENTITY % limit.qname "%MATHML.pfx;limit" >
<!ENTITY % list.qname "%MATHML.pfx;list" >
<!ENTITY % ln.qname "%MATHML.pfx;ln" >
<!ENTITY % log.qname "%MATHML.pfx;log" >
<!ENTITY % logbase.qname "%MATHML.pfx;logbase" >
<!ENTITY % lowlimit.qname "%MATHML.pfx;lowlimit" >
<!ENTITY % lt.qname "%MATHML.pfx;lt" >
<!ENTITY % maction.qname "%MATHML.pfx;maction" >
<!ENTITY % maligngroup.qname "%MATHML.pfx;maligngroup" >
<!ENTITY % malignmark.qname "%MATHML.pfx;malignmark" >

<!ENTITY % math.qname "%MATHML.pfx;math" >
<!ENTITY % matrix.qname "%MATHML.pfx;matrix" >
<!ENTITY % matrixrow.qname "%MATHML.pfx;matrixrow" >
<!ENTITY % max.qname "%MATHML.pfx;max" >
<!ENTITY % mean.qname "%MATHML.pfx;mean" >
<!ENTITY % median.qname "%MATHML.pfx;median" >
<!ENTITY % menclose.qname "%MATHML.pfx;menclose" >
<!ENTITY % merror.qname "%MATHML.pfx;merror" >
<!ENTITY % mfenced.qname "%MATHML.pfx;mfenced" >
<!ENTITY % mfrac.qname "%MATHML.pfx;mfrac" >
<!ENTITY % mglyph.qname "%MATHML.pfx;mglyph" >
<!ENTITY % mi.qname "%MATHML.pfx;mi" >
<!ENTITY % min.qname "%MATHML.pfx;min" >
<!ENTITY % minus.qname "%MATHML.pfx;minus" >
<!ENTITY % mlabeledtr.qname "%MATHML.pfx;mlabeledtr" >
<!ENTITY % mlongdiv.qname "%MATHML.pfx;mlongdiv" >
<!ENTITY % mmultiscripts.qname "%MATHML.pfx;mmultiscripts" >
<!ENTITY % mn.qname "%MATHML.pfx;mn" >
<!ENTITY % mo.qname "%MATHML.pfx;mo" >
<!ENTITY % mode.qname "%MATHML.pfx;mode" >
<!ENTITY % moment.qname "%MATHML.pfx;moment" >
<!ENTITY % momentabout.qname "%MATHML.pfx;momentabout" >
<!ENTITY % mover.qname "%MATHML.pfx;mover" >
<!ENTITY % mpadded.qname "%MATHML.pfx;mpadded" >
<!ENTITY % mphantom.qname "%MATHML.pfx;mphantom" >
<!ENTITY % mprescripts.qname "%MATHML.pfx;mprescripts" >
<!ENTITY % mroot.qname "%MATHML.pfx;mroot" >
<!ENTITY % mrow.qname "%MATHML.pfx;mrow" >
<!ENTITY % ms.qname "%MATHML.pfx;ms" >
<!ENTITY % mscarries.qname "%MATHML.pfx;mscarries" >
<!ENTITY % mscarry.qname "%MATHML.pfx;mscarry" >
<!ENTITY % msgroup.qname "%MATHML.pfx;msgroup" >
<!ENTITY % msline.qname "%MATHML.pfx;msline" >
<!ENTITY % mspace.qname "%MATHML.pfx;mspace" >
<!ENTITY % msqrt.qname "%MATHML.pfx;msqrt" >
<!ENTITY % msrow.qname "%MATHML.pfx;msrow" >
<!ENTITY % mstack.qname "%MATHML.pfx;mstack" >
<!ENTITY % mstyle.qname "%MATHML.pfx;mstyle" >
<!ENTITY % msub.qname "%MATHML.pfx;msub" >
<!ENTITY % msubsup.qname "%MATHML.pfx;msubsup" >
<!ENTITY % msup.qname "%MATHML.pfx;msup" >
<!ENTITY % mtable.qname "%MATHML.pfx;mtable" >
<!ENTITY % mtd.qname "%MATHML.pfx;mtd" >
<!ENTITY % mtext.qname "%MATHML.pfx;mtext" >
<!ENTITY % mtr.qname "%MATHML.pfx;mtr" >
<!ENTITY % munder.qname "%MATHML.pfx;munder" >
<!ENTITY % munderover.qname "%MATHML.pfx;munderover" >
<!ENTITY % naturalnumbers.qname "%MATHML.pfx;naturalnumbers" >

<!ENTITY % neq.qname "%MATHML.pfx;neq" >
<!ENTITY % none.qname "%MATHML.pfx;none" >
<!ENTITY % not.qname "%MATHML.pfx;not" >
<!ENTITY % notanumber.qname "%MATHML.pfx;notanumber" >
<!ENTITY % notin.qname "%MATHML.pfx;notin" >
<!ENTITY % notprsubset.qname "%MATHML.pfx;notprsubset" >
<!ENTITY % notsubset.qname "%MATHML.pfx;notsubset" >
<!ENTITY % or.qname "%MATHML.pfx;or" >
<!ENTITY % otherwise.qname "%MATHML.pfx;otherwise" >
<!ENTITY % outerproduct.qname "%MATHML.pfx;outerproduct" >
<!ENTITY % partialdiff.qname "%MATHML.pfx;partialdiff" >
<!ENTITY % pi.qname "%MATHML.pfx;pi" >
<!ENTITY % piece.qname "%MATHML.pfx;piece" >
<!ENTITY % piecewise.qname "%MATHML.pfx;piecewise" >
<!ENTITY % plus.qname "%MATHML.pfx;plus" >
<!ENTITY % power.qname "%MATHML.pfx;power" >
<!ENTITY % primes.qname "%MATHML.pfx;primes" >
<!ENTITY % product.qname "%MATHML.pfx;product" >
<!ENTITY % prsubset.qname "%MATHML.pfx;prsubset" >
<!ENTITY % quotient.qname "%MATHML.pfx;quotient" >
<!ENTITY % rationals.qname "%MATHML.pfx;rationals" >
<!ENTITY % real.qname "%MATHML.pfx;real" >
<!ENTITY % reals.qname "%MATHML.pfx;reals" >
<!ENTITY % reln.qname "%MATHML.pfx;reln" >
<!ENTITY % rem.qname "%MATHML.pfx;rem" >
<!ENTITY % root.qname "%MATHML.pfx;root" >
<!ENTITY % scalarproduct.qname "%MATHML.pfx;scalarproduct" >
<!ENTITY % sdev.qname "%MATHML.pfx;sdev" >
<!ENTITY % sec.qname "%MATHML.pfx;sec" >
<!ENTITY % sech.qname "%MATHML.pfx;sech" >
<!ENTITY % selector.qname "%MATHML.pfx;selector" >
<!ENTITY % semantics.qname "%MATHML.pfx;semantics" >
<!ENTITY % sep.qname "%MATHML.pfx;sep" >
<!ENTITY % set.qname "%MATHML.pfx;set" >
<!ENTITY % setdiff.qname "%MATHML.pfx;setdiff" >
<!ENTITY % share.qname "%MATHML.pfx;share" >
<!ENTITY % sin.qname "%MATHML.pfx;sin" >
<!ENTITY % sinh.qname "%MATHML.pfx;sinh" >
<!ENTITY % subset.qname "%MATHML.pfx;subset" >
<!ENTITY % sum.qname "%MATHML.pfx;sum" >
<!ENTITY % tan.qname "%MATHML.pfx;tan" >
<!ENTITY % tanh.qname "%MATHML.pfx;tanh" >
<!ENTITY % tendsto.qname "%MATHML.pfx;tendsto" >
<!ENTITY % times.qname "%MATHML.pfx;times" >
<!ENTITY % transpose.qname "%MATHML.pfx;transpose" >
<!ENTITY % true.qname "%MATHML.pfx;true" >
<!ENTITY % union.qname "%MATHML.pfx;union" >
<!ENTITY % uplimit.qname "%MATHML.pfx;uplimit" >

```
<!ENTITY % variance.qname "%MATHML.pfx;variance" >
<!ENTITY % vector.qname "%MATHML.pfx;vector" >
<!ENTITY % vectorproduct.qname "%MATHML.pfx;vectorproduct" >
<!ENTITY % xor.qname "%MATHML.pfx;xor" >

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/mathml/mathml3/mathml3-qname.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- XHTML Document Scripting Module  ..................................... -->
<!-- file: xhtml-script-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-script-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Scripting 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-script-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Scripting

      script, noscript

   This module declares element types and attributes used to provide
   support for executable scripts as well as an alternate content
   container where scripts are not supported.
-->

<!-- script: Scripting Statement ..................... -->

<!ENTITY % script.element  "INCLUDE" >
<![%script.element;[
<!ENTITY % script.content  "( #PCDATA )" >
<!ENTITY % script.qname  "script" >
<!ELEMENT %script.qname;  %script.content; >
<!-- end of script.element -->]]>

<!ENTITY % script.attlist  "INCLUDE" >
<![%script.attlist;[
<!ATTLIST %script.qname;
```

```
    %XHTML.xmlns.attrib;
   %id.attrib;
    xml:space    ( preserve )            #FIXED 'preserve'
    charset      %Charset.datatype;      #IMPLIED
    type         %ContentType.datatype;  #REQUIRED
    src          %URI.datatype;          #IMPLIED
    defer        ( defer )               #IMPLIED
>
<!-- end of script.attlist --]]>

<!-- noscript: No-Script Alternate Content ..... -->

<!ENTITY % noscript.element  "INCLUDE" >
<![%noscript.element;[
<!ENTITY % noscript.content
   "( %Block.mix; )+"
>
<!ENTITY % noscript.qname  "noscript" >
<!ELEMENT %noscript.qname;  %noscript.content; >
<!-- end of noscript.element --]]>

<!ENTITY % noscript.attlist  "INCLUDE" >
<![%noscript.attlist;[
<!ATTLIST %noscript.qname;
    %Common.attrib;
>
<!-- end of noscript.attlist --]]>

<!-- end of xhtml-script-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-script-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- SVG 1.1 Pattern Module ..................................................... -->
<!-- file: svg-pattern.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-pattern.mod,v 1.4 2002/11/14 15:11:03 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Pattern//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-pattern.mod"
```

```
    ..... -->

<!-- Pattern

      pattern

   This module declares markup to provide support for pattern fill.
-->

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.pattern.qname "pattern" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Pattern.class ..... -->

<!ENTITY % SVG.Pattern.extra.class "" >

<!ENTITY % SVG.Pattern.class
   "| %SVG.pattern.qname; %SVG.Pattern.extra.class;"
>

<!-- SVG.Presentation.attrib ..... -->
```

```
<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
   %SVG.Viewport.attrib;
   %SVG.Text.attrib;
   %SVG.TextContent.attrib;
   %SVG.Font.attrib;
   %SVG.Paint.attrib;
   %SVG.Color.attrib;
   %SVG.Opacity.attrib;
   %SVG.Graphics.attrib;
   %SVG.Marker.attrib;
   %SVG.ColorProfile.attrib;
   %SVG.Gradient.attrib;
   %SVG.Clip.attrib;
   %SVG.Mask.attrib;
   %SVG.Filter.attrib;
   %SVG.FilterColor.attrib;
   %SVG.Cursor.attrib;
   flood-color %SVGColor.datatype; #IMPLIED
   flood-opacity %OpacityValue.datatype; #IMPLIED
   lighting-color %SVGColor.datatype; #IMPLIED
   %SVG.Presentation.extra.attrib;"
>

<!-- pattern: Pattern Element ..... -->

<!ENTITY % SVG.pattern.extra.content "" >

<!ENTITY % SVG.pattern.element "INCLUDE" >
<![%SVG.pattern.element;[
<!ENTITY % SVG.pattern.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
     %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
     %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
     %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
     %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
     %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
     %SVG.Font.class; %SVG.pattern.extra.content; )*"
>
<!ELEMENT %SVG.pattern.qname; %SVG.pattern.content; >
<!-- end of SVG.pattern.element -->]]>

<!ENTITY % SVG.pattern.attlist "INCLUDE" >
<![%SVG.pattern.attlist;[
<!ATTLIST %SVG.pattern.qname;
  %SVG.Core.attrib;
```

```
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.XLink.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #IMPLIED
  height %Length.datatype; #IMPLIED
  patternUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
  patternContentUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
  patternTransform %TransformList.datatype; #IMPLIED
  viewBox %ViewBoxSpec.datatype; #IMPLIED
  preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
>
<!-- end of SVG.pattern.attlist -->]]>

<!-- end of svg-pattern.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-pattern.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Block Structural Module  ....................................... -->
<!-- file: xhtml-blkstruct-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-blkstruct-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Block Structural 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-blkstruct-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Block Structural

      div, p

   This module declares the elements and their attributes used to
   support block-level structural markup.
-->
```

```
<!ENTITY % div.element  "INCLUDE" >
<![%div.element;[
<!ENTITY % div.content
   "( #PCDATA | %Flow.mix; )*"
>
<!ENTITY % div.qname  "div" >
<!ELEMENT %div.qname;  %div.content; >
<!-- end of div.element -->]]>

<!ENTITY % div.attlist  "INCLUDE" >
<![%div.attlist;[
<!-- end of div.attlist -->]]>
<!ATTLIST %div.qname;
    %Common.attrib;
>

<!ENTITY % p.element  "INCLUDE" >
<![%p.element;[
<!ENTITY % p.content
   "( #PCDATA | %Inline.mix; )*" >
<!ENTITY % p.qname  "p" >
<!ELEMENT %p.qname;  %p.content; >
<!-- end of p.element -->]]>

<!ENTITY % p.attlist  "INCLUDE" >
<![%p.attlist;[
<!ATTLIST %p.qname;
    %Common.attrib;
>
<!-- end of p.attlist -->]]>

<!-- end of xhtml-blkstruct-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-blkstruct-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Hyperlinking Module ..................................... -->
<!-- file: svg-hyperlink.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-hyperlink.mod,v 1.4 2002/11/14 15:11:02 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:
```

```
      PUBLIC "-//W3C//ELEMENTS SVG 1.1 Hyperlinking//EN"
      SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-hyperlink.mod"

    ..................................................................... -->

<!-- Hyperlinking

      a

    This module declares markup to provide support for hyper linking.
-->

<!-- link to this target -->
<!ENTITY % LinkTarget.datatype "NMTOKEN" >

<!-- Qualified Names (Default) ....................... -->

<!ENTITY % SVG.a.qname "a" >

<!-- Attribute Collections (Default) ..................... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLinkReplace.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Hyperlink.class ................................ -->

<!ENTITY % SVG.Hyperlink.extra.class "" >
```

```
<!ENTITY % SVG.Hyperlink.class
  "| %SVG.a.qname; %SVG.Hyperlink.extra.class;"
>

<!-- SVG.Presentation.attrib ..... -->

<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
   %SVG.Viewport.attrib;
   %SVG.Text.attrib;
   %SVG.TextContent.attrib;
   %SVG.Font.attrib;
   %SVG.Paint.attrib;
   %SVG.Color.attrib;
   %SVG.Opacity.attrib;
   %SVG.Graphics.attrib;
   %SVG.Marker.attrib;
   %SVG.ColorProfile.attrib;
   %SVG.Gradient.attrib;
   %SVG.Clip.attrib;
   %SVG.Mask.attrib;
   %SVG.Filter.attrib;
   %SVG.FilterColor.attrib;
   %SVG.Cursor.attrib;
   flood-color %SVGColor.datatype; #IMPLIED
   flood-opacity %OpacityValue.datatype; #IMPLIED
   lighting-color %SVGColor.datatype; #IMPLIED
   %SVG.Presentation.extra.attrib;"
>

<!-- a: Anchor Element ..... -->

<!ENTITY % SVG.a.extra.content "" >

<!ENTITY % SVG.a.element "INCLUDE" >
<![%SVG.a.element;[
<!ENTITY % SVG.a.content
  "( #PCDATA | %SVG.Description.class; | %SVG.Animation.class;
     %SVG.Structure.class; %SVG.Conditional.class; %SVG.Image.class;
     %SVG.Style.class; %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
     %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
     %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
     %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
     %SVG.Font.class; %SVG.a.extra.content; )*"
>
<!ELEMENT %SVG.a.qname; %SVG.a.content; >
```

```
<!-- end of SVG.a.element -->]]>

<!ENTITY % SVG.a.attlist "INCLUDE" >
<![%SVG.a.attlist;[
<!ATTLIST %SVG.a.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.XLinkReplace.attrib;
  %SVG.External.attrib;
  transform %TransformList.datatype; #IMPLIED
  target %LinkTarget.datatype; #IMPLIED
>
<!-- end of SVG.a.attlist -->]]>

<!-- end of svg-hyperlink.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-hyperlink.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Target Module  ..... -->
<!-- file: xhtml-target-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-target-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Target 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-target-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Target

      target

   This module declares the 'target' attribute used for opening windows
-->
```

```
<!-- render in this frame -->
<!ENTITY % FrameTarget.datatype "CDATA" >

<!-- add 'target' attribute to 'a' element -->
<!ATTLIST %a.qname;
    target      %FrameTarget.datatype;   #IMPLIED
>

<!-- add 'target' attribute to 'area' element -->
<!ATTLIST %area.qname;
    target      %FrameTarget.datatype;   #IMPLIED
>

<!-- add 'target' attribute to 'link' element -->
<!ATTLIST %link.qname;
    target      %FrameTarget.datatype;   #IMPLIED
>

<!-- add 'target' attribute to 'form' element -->
<!ATTLIST %form.qname;
    target      %FrameTarget.datatype;   #IMPLIED
>

<!-- add 'target' attribute to 'base' element -->
<!ATTLIST %base.qname;
    target      %FrameTarget.datatype;   #IMPLIED
>

<!-- end of xhtml-target-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-target-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Images Module  ................................................. -->
<!-- file: xhtml-image-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Rovision: $Id: xhtml-image-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Images 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-image-1.mod"
```

```
    Revisions:
    (none)
    ..... -->

<!-- Images

      img

    This module provides markup to support basic image embedding.
-->

<!-- To avoid problems with text-only UAs as well as to make
    image content understandable and navigable to users of
    non-visual UAs, you need to provide a description with
    the 'alt' attribute, and avoid server-side image maps.
-->

<!ENTITY % img.element  "INCLUDE" >
<![%img.element;[
<!ENTITY % img.content  "EMPTY" >
<!ENTITY % img.qname  "img" >
<!ELEMENT %img.qname;  %img.content; >
<!-- end of img.element -->]]>

<!ENTITY % img.attlist  "INCLUDE" >
<![%img.attlist;[
<!ATTLIST %img.qname;
    %Common.attrib;
    src         %URI.datatype;          #REQUIRED
    alt         %Text.datatype;         #REQUIRED
    longdesc    %URI.datatype;          #IMPLIED
    height      %Length.datatype;       #IMPLIED
    width       %Length.datatype;       #IMPLIED
>
<!-- end of img.attlist -->]]>

<!-- end of xhtml-image-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-image-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Datatypes Module  ..... -->
<!-- file: xhtml-datatypes-1.mod

    This is XHTML, a reformulation of HTML as a modular XML application.
```

Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
Revision: $Id: xhtml-datatypes-1.mod,v 4.1 2001/04/06 19:23:32 altheim Exp $ SMI

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES XHTML Datatypes 1.0//EN"
SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-datatypes-1.mod"

Revisions:
(none)
..... -->

<!-- Datatypes

defines containers for the following datatypes, many of
these imported from other specifications and standards.
-->

<!-- Length defined for cellpadding/cellspacing -->

<!-- nn for pixels or nn% for percentage length -->
<!ENTITY % Length.datatype "CDATA" >

<!-- space-separated list of link types -->
<!ENTITY % LinkTypes.datatype "NMTOKENS" >

<!-- single or comma-separated list of media descriptors -->
<!ENTITY % MediaDesc.datatype "CDATA" >

<!-- pixel, percentage, or relative -->
<!ENTITY % MultiLength.datatype "CDATA" >

<!-- one or more digits (NUMBER) -->
<!ENTITY % Number.datatype "CDATA" >

<!-- integer representing length in pixels -->
<!ENTITY % Pixels.datatype "CDATA" >

<!-- script expression -->
<!ENTITY % Script.datatype "CDATA" >

<!-- textual content -->
<!ENTITY % Text.datatype "CDATA" >

<!-- Placeholder Compact URI-related types -->
<!ENTITY % CURIE.datatype "CDATA" >
<!ENTITY % CURIEs.datatype "CDATA" >
<!ENTITY % SafeCURIE.datatype "CDATA" >

```
<!ENTITY % SafeCURIEs.datatype "CDATA" >
<!ENTITY % URIorSafeCURIE.datatype "CDATA" >
<!ENTITY % URIorSafeCURIEs.datatype "CDATA" >

<!-- Imported Datatypes ..... -->

<!-- a single character from [ISO10646] -->
<!ENTITY % Character.datatype "CDATA" >

<!-- a character encoding, as per [RFC2045] -->
<!ENTITY % Charset.datatype "CDATA" >

<!-- a space separated list of character encodings, as per [RFC2045] -->
<!ENTITY % Charsets.datatype "CDATA" >

<!-- Color specification using color name or sRGB (#RRGGBB) values -->
<!ENTITY % Color.datatype "CDATA" >

<!-- media type, as per [RFC2045] -->
<!ENTITY % ContentType.datatype "CDATA" >

<!-- comma-separated list of media types, as per [RFC2045] -->
<!ENTITY % ContentTypes.datatype "CDATA" >

<!-- date and time information. ISO date format -->
<!ENTITY % Datetime.datatype "CDATA" >

<!-- formal public identifier, as per [ISO8879] -->
<!ENTITY % FPI.datatype "CDATA" >

<!-- a language code, as per [RFC3066] or its successor -->
<!ENTITY % LanguageCode.datatype "CDATA" >

<!-- a comma separated list of language code ranges -->
<!ENTITY % LanguageCodes.datatype "CDATA" >

<!-- a qualified name , as per [XMLNS] or its successor -->
<!ENTITY % QName.datatype "CDATA" >
<!ENTITY % QNames.datatype "CDATA" >

<!-- a Uniform Resource Identifier, see [URI] -->
<!ENTITY % URI.datatype "CDATA" >

<!-- a space-separated list of Uniform Resource Identifiers, see [URI] -->
<!ENTITY % URIs.datatype "CDATA" >

<!-- end of xhtml-datatypes-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-datatypes-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG Template Qualified Name Module ..................................... -->
<!-- file: svg-template-qname.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-template-qname.mod,v 1.1 2002/04/20 18:08:11 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG Template Qualified Name//EN"
     SYSTEM "svg-template-qname.mod"

   ..................................................................... -->

<!-- Template Qualified Name

   This module is contained in two parts, labeled Section 'A' and 'B':

     Section A declares parameter entities to support namespace-
     qualified names, namespace declarations, and name prefixing
     for SVG and extensions.

     Section B declares parameter entities used to provide
     namespace-qualified names for all SVG element types:
-->

<!-- Section A: Template XML Namespace Framework :::::::::::::::::: -->

<!-- 1. Declare a %MODULE.prefixed; conditional section keyword, used
     to activate namespace prefixing. The default value should
     inherit '%NS.prefixed;' from the DTD driver, so that unless
     overridden, the default behaviour follows the overall DTD
     prefixing scheme.
-->
<!ENTITY % NS.prefixed "IGNORE" >
<!ENTITY % MODULE.prefixed "%NS.prefixed;" >

<!-- 2. Declare a parameter entity (eg., %MODULE.xmlns;) containing
     the URI reference used to identify the Template namespace:
-->
<!ENTITY % MODULE.xmlns "http://template-namespace-uri" >
```

```
<!-- 3. Declare parameter entities (eg., %MODULE.prefix;) containing
      the default namespace prefix string(s) to use when prefixing
      is enabled. This may be overridden in the DTD driver or the
      internal subset of an document instance. If no default prefix
      is desired, this may be declared as an empty string.
-->
<!ENTITY % MODULE.prefix "" >

<!-- 4. Declare parameter entities (eg., %MODULE.pfx;) containing the
      colonized prefix(es) (eg., '%MODULE.prefix;:') used when
      prefixing is active, an empty string when it is not.
-->
<![%MODULE.prefixed;[
<!ENTITY % MODULE.pfx "%MODULE.prefix;:" >
]]>
<!ENTITY % MODULE.pfx "" >

<!-- 5. The parameter entity %MODULE.xmlns.extra.attrib; may be
      redeclared to contain any non-Template namespace declaration
      attributes for namespaces embedded in SVG. The default
      is an empty string.
-->
<![%MODULE.prefixed;[
<!ENTITY % MODULE.xmlns.extra.attrib
  "xmlns:%MODULE.prefix; %URI.datatype; #FIXED '%MODULE.xmlns;'" >
]]>
<!ENTITY % MODULE.xmlns.extra.attrib "" >

<!ENTITY % SVG.xmlns.extra.attrib
  "%MODULE.xmlns.extra.attrib;"
>

<!-- Section B: SVG Qualified Names ::::::::::::::::::::::::::::: -->

<!-- 6. This section declares parameter entities used to provide
      namespace-qualified names for all Template element types.
-->

<!-- module: svg-template.mod ..... -->

<!ENTITY % MODULE.element.qname "%MODULE.pfx;element" >

<!-- end of svg-template-qname.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/templates/svg-template.mod
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
```

jar/net/sf/saxon/data/w3c/svg11/templates/svg-template-qname.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Modular Framework Module ..................................... -->
<!-- file: svg-framework.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-framework.mod,v 1.2 2002/04/20 18:07:43 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Modular Framework//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-framework.mod"

   ..................................................................... -->

<!-- Modular Framework

   This module instantiates the modules needed o support the SVG
   modularization model, including:

     + Datatypes
     + Qualified Name
     + Document Model
     + Attribute Collection
-->

<!ENTITY % svg-datatypes.module "INCLUDE" >
<![%svg-datatypes.module;[
<!ENTITY % svg-datatypes.mod
  PUBLIC "-//W3C//ENTITIES SVG 1.1 Datatypes//EN"
       "svg-datatypes.mod" >
%svg-datatypes.mod;]]>

<!ENTITY % svg-qname.module "INCLUDE" >
<![%svg-qname.module;[
<!ENTITY % svg-qname.mod
  PUBLIC "-//W3C//ENTITIES SVG 1.1 Qualified Name//EN"
       "svg-qname.mod" >
%svg-qname.mod;]]>

<!ENTITY % svg-model.module "INCLUDE" >
<![%svg-model.module;[
<!-- instantiate the Document Model declared in the DTD driver -->
%svg-model.mod;]]>
```

<!ENTITY % svg-attribs.module "INCLUDE" >
<![%svg-attribs.module;[
<!-- instantiate the Attribute Collection declared in the DTD driver -->
%svg-attribs.mod;]]>

<!-- end of svg-framework.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-framework.mod
No license file was found, but licenses were detected in source scan.

2001 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the XHTML DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Murray M. Altheim <altheim@eng.sun.com>
Revision: $Id: xhtml11.dtd,v 1.21 2001/05/29 16:37:01 ahby Exp $

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml11.dtd
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml11.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- RDDL 1.0 Document Model Module ..... -->
<!-- file: xhtml-rddl-model-1.mod

Derived from:
-->
<!--

This is XHTML Basic, a proper subset of XHTML.
Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: xhtml-rddl-model-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

```
     PUBLIC "-//XML-DEV//ENTITIES RDDL Document Model 1.0//EN"
     SYSTEM "http://www.rddl.org/xhtml-rddl-model-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- RDDL Document Model

   This module describes the groupings of elements that make up
   common content models for XHTML elements.
-->

<!-- Optional Elements in head  ..... -->

<!ENTITY % HeadOpts.mix
   "( %meta.qname; | %link.qname; | %object.qname; )*" >

<!-- Miscellaneous Elements  ..... -->

<!ENTITY % Misc.class "" >

<!-- Inline Elements  ..... -->

<!ENTITY % InlStruct.class "%br.qname; | %span.qname;" >

<!ENTITY % InlPhras.class
   "| %em.qname; | %strong.qname; | %dfn.qname; | %code.qname;
    | %samp.qname; | %kbd.qname; | %var.qname; | %cite.qname;
    | %abbr.qname; | %acronym.qname; | %q.qname;" >

<!ENTITY % InlPres.class "" >

<!ENTITY % I18n.class "" >

<!ENTITY % Anchor.class "| %a.qname;" >

<!ENTITY % InlSpecial.class "| %img.qname; | %object.qname;" >

<!ENTITY % InlForm.class
   "| %input.qname; | %select.qname; | %textarea.qname;
    | %label.qname;"
>

<!ENTITY % Inline.extra "" >

<!ENTITY % Inline.class
```

```
    "%InlStruct.class;
     %InlPhras.class;
     %Anchor.class;
     %InlSpecial.class;
     %InlForm.class;
     %Inline.extra;"
>

<!ENTITY % InlNoAnchor.class
    "%InlStruct.class;
     %InlPhras.class;
     %InlSpecial.class;
     %InlForm.class;
     %Inline.extra;"
>

<!ENTITY % InlNoAnchor.mix
    "%InlNoAnchor.class;
     %Misc.class;"
>

<!ENTITY % Inline.mix
    "%Inline.class;
     %Misc.class;"
>

<!-- Block Elements  ..... -->

<!ENTITY % Heading.class
    "%h1.qname; | %h2.qname; | %h3.qname;
     | %h4.qname; | %h5.qname; | %h6.qname;"
>
<!ENTITY % List.class  "%ul.qname; | %ol.qname; | %dl.qname;" >

<!ENTITY % Table.class "| %table.qname;" >

<!ENTITY % Form.class  "| %form.qname;" >

<!ENTITY % BlkStruct.class "%p.qname; | %div.qname; | %RDDL.resource.qname;" >

<!ENTITY % BlkPhras.class
    "| %pre.qname; | %blockquote.qname; | %address.qname;"
>

<!ENTITY % BlkPres.class "" >

<!ENTITY % BlkSpecial.class
    "%Table.class;
```

```
    %Form.class;"
>

<!ENTITY % Block.extra "" >

<!ENTITY % Block.class
   "%BlkStruct.class;
    %BlkPhras.class;
    %BlkSpecial.class;
    %Block.extra;"
>

<!ENTITY % Block.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    %Misc.class;"
>

<!-- All Content Elements  ..... -->

<!-- declares all content except tables
-->
<!ENTITY % FlowNoTable.mix
   "%Heading.class;
    | %List.class;
    | %BlkStruct.class;
    %BlkPhras.class;
    %Form.class;
    %Block.extra;
    | %Inline.class;
    %Misc.class;"
>

<!ENTITY % Flow.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    | %Inline.class;
    %Misc.class;"
>

<!-- end of xhtml-basic10-model-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-rddl-model-1.mod
```

No license file was found, but licenses were detected in source scan.

```
<!-- ..... -->
<!-- XHTML Text Module  ..... -->
<!-- file: xhtml-text-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-text-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Text 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-text-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Textual Content

   The Text module includes declarations for all core
   text container elements and their attributes.
-->

<!ENTITY % xhtml-inlstruct.module "INCLUDE" >
<![%xhtml-inlstruct.module;[
<!ENTITY % xhtml-inlstruct.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Inline Structural 1.0//EN"
         "xhtml-inlstruct-1.mod" >
%xhtml-inlstruct.mod;]]>

<!ENTITY % xhtml-inlphras.module "INCLUDE" >
<![%xhtml-inlphras.module;[
<!ENTITY % xhtml-inlphras.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Inline Phrasal 1.0//EN"
         "xhtml-inlphras-1.mod" >
%xhtml-inlphras.mod;]]>

<!ENTITY % xhtml-blkstruct.module "INCLUDE" >
<![%xhtml-blkstruct.module;[
<!ENTITY % xhtml-blkstruct.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Block Structural 1.0//EN"
         "xhtml-blkstruct-1.mod" >
%xhtml-blkstruct.mod;]]>

<!ENTITY % xhtml-blkphras.module "INCLUDE" >
<![%xhtml-blkphras.module;[
```

```
<!ENTITY % xhtml-blkphras.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Block Phrasal 1.0//EN"
         "xhtml-blkphras-1.mod" >
%xhtml-blkphras.mod;]]>

<!-- end of xhtml-text-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-text-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 External Resources Attribute Module ..... -->
<!-- file: svg-extresources-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-extresources-attrib.mod,v 1.2 2002/04/20 18:07:42 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 External Resources Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-extresources-attrib.mod"

   ..................................................................... -->

<!-- External Resources Attribute

     externalResourcesRequired

   This module defines the External attribute set.
-->

<!ENTITY % SVG.externalResourcesRequired.attrib
  "externalResourcesRequired %Boolean.datatype; #IMPLIED"
>

<!ENTITY % SVG.External.extra.attrib "" >

<!ENTITY % SVG.External.attrib
  "%SVG.externalResourcesRequired.attrib;
   %SVG.External.extra.attrib;"
>

<!-- end of svg-extresources-attrib.mod -->

Found in path(s):
```

* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-extresources-attrib.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Presentation Module ..................................... -->
<!-- file: xhtml-pres-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-pres-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Presentation 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-pres-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Presentational Elements

   This module defines elements and their attributes for
   simple presentation-related markup.
-->

<!ENTITY % xhtml-inlpres.module "INCLUDE" >
<![%xhtml-inlpres.module;[
<!ENTITY % xhtml-inlpres.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Inline Presentation 1.0//EN"
         "xhtml-inlpres-1.mod" >
%xhtml-inlpres.mod;]]>

<!ENTITY % xhtml-blkpres.module "INCLUDE" >
<![%xhtml-blkpres.module;[
<!ENTITY % xhtml-blkpres.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Block Presentation 1.0//EN"
         "xhtml-blkpres-1.mod" >
%xhtml-blkpres.mod;]]>

<!-- end of xhtml-pres-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-pres-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Paint Opacity Attribute Module ..................................... -->
<!-- file: svg-opacity-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-opacity-attrib.mod,v 1.2 2002/04/20 18:07:43 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Paint Opacity Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-opacity-attrib.mod"

   ..................................................................... -->

<!-- Paint Opacity Attribute

      opacity, fill-opacity, stroke-opacity

   This module defines the Opacity attribute set.
-->

<!ENTITY % SVG.opacity.attrib
  "opacity %OpacityValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.fill-opacity.attrib
  "fill-opacity %OpacityValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-opacity.attrib
  "stroke-opacity %OpacityValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.Opacity.extra.attrib "" >

<!ENTITY % SVG.Opacity.attrib
  "%SVG.opacity.attrib;
   %SVG.fill-opacity.attrib;
   %SVG.stroke-opacity.attrib;
   %SVG.Opacity.extra.attrib;"
>

<!-- end of svg-opacity-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-opacity-attrib.mod
```

No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Basic Structure Module ....................................... -->
<!-- file: svg-basic-structure.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-basic-structure.mod,v 1.6 2002/11/14 15:11:02 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Basic Structure//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-structure.mod"

   ..................................................................... -->

<!-- Basic Structure

     svg, g, defs, desc, title, metadata, use

   This module declares the major structural elements and their attributes.
-->

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.svg.qname "svg" >
<!ENTITY % SVG.g.qname "g" >
<!ENTITY % SVG.defs.qname "defs" >
<!ENTITY % SVG.desc.qname "desc" >
<!ENTITY % SVG.title.qname "title" >
<!ENTITY % SVG.metadata.qname "metadata" >
<!ENTITY % SVG.use.qname "use" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
```

```
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.DocumentEvents.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLinkEmbed.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Description.class ..... -->

<!ENTITY % SVG.Description.extra.class "" >

<!ENTITY % SVG.Description.class
  "%SVG.desc.qname; | %SVG.title.qname; | %SVG.metadata.qname;
   %SVG.Description.extra.class;"
>

<!-- SVG.Use.class ..... -->

<!ENTITY % SVG.Use.extra.class "" >

<!ENTITY % SVG.Use.class
  "| %SVG.use.qname; %SVG.Use.extra.class;"
>

<!-- SVG.Structure.class ..... -->

<!ENTITY % SVG.Structure.extra.class "" >

<!ENTITY % SVG.Structure.class
  "| %SVG.g.qname; | %SVG.defs.qname; %SVG.Use.class;
     %SVG.Structure.extra.class;"
>

<!-- SVG.Presentation.attrib ..... -->

<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
   %SVG.Viewport.attrib;
   %SVG.Text.attrib;
   %SVG.TextContent.attrib;
```

```
    %SVG.Font.attrib;
    %SVG.Paint.attrib;
    %SVG.Color.attrib;
    %SVG.Opacity.attrib;
    %SVG.Graphics.attrib;
    %SVG.Marker.attrib;
    %SVG.ColorProfile.attrib;
    %SVG.Gradient.attrib;
    %SVG.Clip.attrib;
    %SVG.Mask.attrib;
    %SVG.Filter.attrib;
    %SVG.FilterColor.attrib;
    %SVG.Cursor.attrib;
    flood-color %SVGColor.datatype; #IMPLIED
    flood-opacity %OpacityValue.datatype; #IMPLIED
    lighting-color %SVGColor.datatype; #IMPLIED
    %SVG.Presentation.extra.attrib;"
>

<!-- svg: SVG Document Element ..... -->

<!ENTITY % SVG.svg.extra.content "" >

<!ENTITY % SVG.svg.element "INCLUDE" >
<![%SVG.svg.element;[
<!ENTITY % SVG.svg.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
     %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
     %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
     %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
     %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
     %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
     %SVG.Font.class; %SVG.svg.extra.content; )*"
>
<!ELEMENT %SVG.svg.qname; %SVG.svg.content; >
<!-- end of SVG.svg.element -->]]>

<!ENTITY % SVG.svg.attlist "INCLUDE" >
<![%SVG.svg.attlist;[
<!ATTLIST %SVG.svg.qname;
  %SVG.xmlns.attrib;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.DocumentEvents.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.External.attrib;
```

```
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #IMPLIED
  height %Length.datatype; #IMPLIED
  viewBox %ViewBoxSpec.datatype; #IMPLIED
  preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
  zoomAndPan ( disable | magnify ) 'magnify'
  version %Number.datatype; #FIXED '1.1'
  baseProfile %Text.datatype; #IMPLIED
>
<!-- end of SVG.svg.attlist -->]]>

<!-- g: Group Element ..... -->

<!ENTITY % SVG.g.extra.content "" >

<!ENTITY % SVG.g.element "INCLUDE" >
<![%SVG.g.element;[
<!ENTITY % SVG.g.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
    %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
    %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
    %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
    %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
    %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
    %SVG.Font.class; %SVG.g.extra.content; )*"
>
<!ELEMENT %SVG.g.qname; %SVG.g.content; >
<!-- end of SVG.g.element -->]]>

<!ENTITY % SVG.g.attlist "INCLUDE" >
<![%SVG.g.attlist;[
<!ATTLIST %SVG.g.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.External.attrib;
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.g.attlist -->]]>

<!-- defs: Definisions Element ..... -->

<!ENTITY % SVG.defs.extra.content "" >

<!ENTITY % SVG.defs.element "INCLUDE" >
```

```
<![%SVG.defs.element;[
<!ENTITY % SVG.defs.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
    %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
    %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
    %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
    %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
    %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
    %SVG.Font.class; %SVG.defs.extra.content; )*"
>
<!ELEMENT %SVG.defs.qname; %SVG.defs.content; >
<!-- end of SVG.defs.element -->]]>

<!ENTITY % SVG.defs.attlist "INCLUDE" >
<![%SVG.defs.attlist;[
<!ATTLIST %SVG.defs.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.External.attrib;
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.defs.attlist -->]]>

<!-- desc: Description Element ..... -->

<!ENTITY % SVG.desc.extra.content "" >

<!ENTITY % SVG.desc.element "INCLUDE" >
<![%SVG.desc.element;[
<!ENTITY % SVG.desc.content
  "( #PCDATA %SVG.desc.extra.content; )*"
>
<!ELEMENT %SVG.desc.qname; %SVG.desc.content; >
<!-- end of SVG.desc.element -->]]>

<!ENTITY % SVG.desc.attlist "INCLUDE" >
<![%SVG.desc.attlist;[
<!ATTLIST %SVG.desc.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
>
<!-- end of SVG.desc.attlist -->]]>

<!-- title: Title Element ..... -->
```

```
<!ENTITY % SVG.title.extra.content "" >

<!ENTITY % SVG.title.element "INCLUDE" >
<![%SVG.title.element;[
<!ENTITY % SVG.title.content
  "( #PCDATA %SVG.title.extra.content; )*"
>
<!ELEMENT %SVG.title.qname; %SVG.title.content; >
<!-- end of SVG.title.element -->]]>

<!ENTITY % SVG.title.attlist "INCLUDE" >
<![%SVG.title.attlist;[
<!ATTLIST %SVG.title.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
>
<!-- end of SVG.title.attlist -->]]>

<!-- metadata: Metadata Element ..... -->

<!ENTITY % SVG.metadata.extra.content "" >

<!ENTITY % SVG.metadata.element "INCLUDE" >
<![%SVG.metadata.element;[
<!ENTITY % SVG.metadata.content
  "( #PCDATA %SVG.metadata.extra.content; )*"
>
<!ELEMENT %SVG.metadata.qname; %SVG.metadata.content; >
<!-- end of SVG.metadata.element -->]]>

<!ENTITY % SVG.metadata.attlist "INCLUDE" >
<![%SVG.metadata.attlist;[
<!ATTLIST %SVG.metadata.qname;
  %SVG.Core.attrib;
>
<!-- end of SVG.metadata.attlist -->]]>

<!-- use: Use Element ..... -->

<!ENTITY % SVG.use.extra.content "" >

<!ENTITY % SVG.use.element "INCLUDE" >
<![%SVG.use.element;[
<!ENTITY % SVG.use.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class;
     %SVG.use.extra.content; )*)"
>
<!ELEMENT %SVG.use.qname; %SVG.use.content; >
```

```
<!-- end of SVG.use.element -->]]>

<!ENTITY % SVG.use.attlist "INCLUDE" >
<![%SVG.use.attlist;[
<!ATTLIST %SVG.use.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.XLinkEmbed.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #IMPLIED
  height %Length.datatype; #IMPLIED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.use.attlist -->]]>

<!-- end of svg-basic-structure.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-basic-structure.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML-Print 1.0 Document Model Module ..... -->
<!-- file: xhtml-print10-model-1.mod

   This is XHTML-Print 1.0, a variant of XHTML Basic for printing.
   Copyright 1998-2003 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-print10-model-1.mod,v 1.7 2003/10/24 22:14:40 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML-Print 1.0 Document Model 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-print10-model-1.mod"

   ..... -->

<!-- XHTML-Print 1.0 Document Model

   This module describes the groupings of elements that make up
   common content models for XHTML-Print elements.
-->
```

```
<!-- Optional Elements in head ..... -->

<!ENTITY % HeadOpts.mix
   "( %script.qname; | %style.qname; | %meta.qname; | %link.qname;
    | %object.qname; )*" >

<!-- Miscellaneous Elements ..... -->

<!ENTITY % Script.class "| %script.qname; | %noscript.qname;" >

<!ENTITY % Misc.extra "" >

<!ENTITY % Misc.class
   "%Script.class;
    %Misc.extra;"
>

<!-- Inline Elements ..... -->

<!ENTITY % InlStruct.class "%br.qname; | %span.qname;" >

<!ENTITY % InlPhras.class
   "| %em.qname; | %strong.qname; | %dfn.qname; | %code.qname;
    | %samp.qname; | %kbd.qname; | %var.qname; | %cite.qname;
    | %abbr.qname; | %acronym.qname; | %q.qname;" >

<!ENTITY % InlPres.class
   "| %tt.qname; | %i.qname; | %b.qname; | %big.qname;
    | %small.qname; | %sub.qname; | %sup.qname; " >

<!ENTITY % I18n.class "" >

<!ENTITY % Anchor.class "| %a.qname;" >

<!ENTITY % InlSpecial.class "| %img.qname; | %object.qname;" >

<!ENTITY % InlForm.class
   "| %input.qname; | %select.qname; | %textarea.qname;
    | %label.qname;"
>

<!ENTITY % Inline.extra "" >

<!ENTITY % Inline.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %Anchor.class;
```

```
    %InlSpecial.class;
    %InlForm.class;
    %Inline.extra;"
>

<!ENTITY % InlNoAnchor.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %InlSpecial.class;
    %InlForm.class;
    %Inline.extra;"
>

<!ENTITY % InlNoAnchor.mix
   "%InlNoAnchor.class;
    %Misc.class;"
>

<!ENTITY % Inline.mix
   "%Inline.class;
    %Misc.class;"
>

<!-- Block Elements ..................................... -->

<!ENTITY % Heading.class
   "%h1.qname; | %h2.qname; | %h3.qname;
    | %h4.qname; | %h5.qname; | %h6.qname;"
>
<!ENTITY % List.class  "%ul.qname; | %ol.qname; | %dl.qname;" >

<!ENTITY % Table.class "| %table.qname;" >

<!ENTITY % Form.class  "| %form.qname;" >

<!ENTITY % BlkStruct.class "%p.qname; | %div.qname;" >

<!ENTITY % BlkPhras.class
   "| %pre.qname; | %blockquote.qname; | %address.qname;"
>

<!ENTITY % BlkPres.class "| %hr.qname;" >

<!ENTITY % BlkSpecial.class
   "%Table.class;
    %Form.class;"
>
```

```
<!ENTITY % Block.extra "" >

<!ENTITY % Block.class
   "%BlkStruct.class;
    %BlkPhras.class;
    %BlkPres.class;
    %BlkSpecial.class;
    %Block.extra;"
>

<!ENTITY % Block.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    %Misc.class;"
>

<!-- All Content Elements ..... -->

<!ENTITY % FlowNoTable.mix
   "%Heading.class;
    | %List.class;
    | %BlkStruct.class;
    %BlkPhras.class;
    %BlkPres.class;
    %Form.class;
    %Block.extra;
    | %Inline.class;
    %Misc.class;"
>

<!ENTITY % Flow.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    | %Inline.class;
    %Misc.class;"
>

<!-- end of xhtml-print10-model-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-print10-model-1.mod
No license file was found, but licenses were detected in source scan.

SVG 1.1 Tiny DTD

This is SVG Tiny, a proper subset of SVG.

The Scalable Vector Graphics (SVG)
Copyright 2001, 2002 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the SVG DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11
dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg11-tiny-flat.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Images Module ..... -->
<!-- file: xhtml-image-1.mod

This is XHTML, a reformulation of HTML as a modular XML application.
Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
Rovision: $Id: xhtml-image-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS XHTML Images 1.0//EN"
SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-image-1.mod"

Revisions:
(none)
..... -->

<!-- Images

img

```
   This module provides markup to support basic image embedding.
-->

<!-- To avoid problems with text-only UAs as well as to make
   image content understandable and navigable to users of
   non-visual UAs, you need to provide a description with
   the 'alt' attribute, and avoid server-side image maps.
-->

<!ENTITY % img.element  "INCLUDE" >
<![%img.element;[
<!ENTITY % img.content  "EMPTY" >
<!ENTITY % img.qname  "img" >
<!ELEMENT %img.qname;  %img.content; >
<!-- end of img.element -->]]>

<!ENTITY % img.attlist  "INCLUDE" >
<![%img.attlist;[
<!ATTLIST %img.qname;
    %Common.attrib;
    src         %URI.datatype;          #REQUIRED
    alt         %Text.datatype;         #REQUIRED
    longdesc    %URI.datatype;          #IMPLIED
    height      %Length.datatype;       #IMPLIED
    width       %Length.datatype;       #IMPLIED
>
<!-- end of img.attlist -->]]>

<!-- end of xhtml-image-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-image-1.mod
No license file was found, but licenses were detected in source scan.

<!--
   File isomfrk.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.
```

```
   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

   Entity names in this file are derived from files carrying the
   following notice:

   (C) International Organization for Standardization 1991
   Permission to copy in any form is granted for use with
   conforming SGML systems and applications as defined in
   ISO 8879, provided this notice is included in all copies.

-->


<!--

     Public identifier: -//W3C//ENTITIES Math Alphabets: Fraktur//EN//XML
     System identifier: http://www.w3.org/2003/entities/2007/isomfrk.ent

   The public identifier should always be used verbatim.
   The system identifier may be changed to suit local requirements.

   Typical invocation:

     <!ENTITY % isomfrk PUBLIC
       "-//W3C//ENTITIES Math Alphabets: Fraktur//EN//XML"
       "http://www.w3.org/2003/entities/2007/isomfrk.ent"
     >
     %isomfrk;

-->

<!ENTITY afr           "&#x1D51E;" ><!--MATHEMATICAL FRAKTUR SMALL A -->
<!ENTITY Afr           "&#x1D504;" ><!--MATHEMATICAL FRAKTUR CAPITAL A -->
<!ENTITY bfr           "&#x1D51F;" ><!--MATHEMATICAL FRAKTUR SMALL B -->
<!ENTITY Bfr           "&#x1D505;" ><!--MATHEMATICAL FRAKTUR CAPITAL B -->
<!ENTITY cfr           "&#x1D520;" ><!--MATHEMATICAL FRAKTUR SMALL C -->
<!ENTITY Cfr           "&#x0212D;" ><!--BLACK-LETTER CAPITAL C -->
<!ENTITY dfr           "&#x1D521;" ><!--MATHEMATICAL FRAKTUR SMALL D -->
<!ENTITY Dfr           "&#x1D507;" ><!--MATHEMATICAL FRAKTUR CAPITAL D -->
<!ENTITY efr           "&#x1D522;" ><!--MATHEMATICAL FRAKTUR SMALL E -->
<!ENTITY Efr           "&#x1D508;" ><!--MATHEMATICAL FRAKTUR CAPITAL E -->
<!ENTITY ffr           "&#x1D523;" ><!--MATHEMATICAL FRAKTUR SMALL F -->
<!ENTITY Ffr           "&#x1D509;" ><!--MATHEMATICAL FRAKTUR CAPITAL F -->
<!ENTITY gfr           "&#x1D524;" ><!--MATHEMATICAL FRAKTUR SMALL G -->
<!ENTITY Gfr           "&#x1D50A;" ><!--MATHEMATICAL FRAKTUR CAPITAL G -->
<!ENTITY hfr           "&#x1D525;" ><!--MATHEMATICAL FRAKTUR SMALL H -->
```

```
<!ENTITY Hfr          "&#x0210C;" ><!--BLACK-LETTER CAPITAL H -->
<!ENTITY ifr          "&#x1D526;" ><!--MATHEMATICAL FRAKTUR SMALL I -->
<!ENTITY Ifr          "&#x02111;" ><!--BLACK-LETTER CAPITAL I -->
<!ENTITY jfr          "&#x1D527;" ><!--MATHEMATICAL FRAKTUR SMALL J -->
<!ENTITY Jfr          "&#x1D50D;" ><!--MATHEMATICAL FRAKTUR CAPITAL J -->
<!ENTITY kfr          "&#x1D528;" ><!--MATHEMATICAL FRAKTUR SMALL K -->
<!ENTITY Kfr          "&#x1D50E;" ><!--MATHEMATICAL FRAKTUR CAPITAL K -->
<!ENTITY lfr          "&#x1D529;" ><!--MATHEMATICAL FRAKTUR SMALL L -->
<!ENTITY Lfr          "&#x1D50F;" ><!--MATHEMATICAL FRAKTUR CAPITAL L -->
<!ENTITY mfr          "&#x1D52A;" ><!--MATHEMATICAL FRAKTUR SMALL M -->
<!ENTITY Mfr          "&#x1D510;" ><!--MATHEMATICAL FRAKTUR CAPITAL M -->
<!ENTITY nfr          "&#x1D52B;" ><!--MATHEMATICAL FRAKTUR SMALL N -->
<!ENTITY Nfr          "&#x1D511;" ><!--MATHEMATICAL FRAKTUR CAPITAL N -->
<!ENTITY ofr          "&#x1D52C;" ><!--MATHEMATICAL FRAKTUR SMALL O -->
<!ENTITY Ofr          "&#x1D512;" ><!--MATHEMATICAL FRAKTUR CAPITAL O -->
<!ENTITY pfr          "&#x1D52D;" ><!--MATHEMATICAL FRAKTUR SMALL P -->
<!ENTITY Pfr          "&#x1D513;" ><!--MATHEMATICAL FRAKTUR CAPITAL P -->
<!ENTITY qfr          "&#x1D52E;" ><!--MATHEMATICAL FRAKTUR SMALL Q -->
<!ENTITY Qfr          "&#x1D514;" ><!--MATHEMATICAL FRAKTUR CAPITAL Q -->
<!ENTITY rfr          "&#x1D52F;" ><!--MATHEMATICAL FRAKTUR SMALL R -->
<!ENTITY Rfr          "&#x0211C;" ><!--BLACK-LETTER CAPITAL R -->
<!ENTITY sfr          "&#x1D530;" ><!--MATHEMATICAL FRAKTUR SMALL S -->
<!ENTITY Sfr          "&#x1D516;" ><!--MATHEMATICAL FRAKTUR CAPITAL S -->
<!ENTITY tfr          "&#x1D531;" ><!--MATHEMATICAL FRAKTUR SMALL T -->
<!ENTITY Tfr          "&#x1D517;" ><!--MATHEMATICAL FRAKTUR CAPITAL T -->
<!ENTITY ufr          "&#x1D532;" ><!--MATHEMATICAL FRAKTUR SMALL U -->
<!ENTITY Ufr          "&#x1D518;" ><!--MATHEMATICAL FRAKTUR CAPITAL U -->
<!ENTITY vfr          "&#x1D533;" ><!--MATHEMATICAL FRAKTUR SMALL V -->
<!ENTITY Vfr          "&#x1D519;" ><!--MATHEMATICAL FRAKTUR CAPITAL V -->
<!ENTITY wfr          "&#x1D534;" ><!--MATHEMATICAL FRAKTUR SMALL W -->
<!ENTITY Wfr          "&#x1D51A;" ><!--MATHEMATICAL FRAKTUR CAPITAL W -->
<!ENTITY xfr          "&#x1D535;" ><!--MATHEMATICAL FRAKTUR SMALL X -->
<!ENTITY Xfr          "&#x1D51B;" ><!--MATHEMATICAL FRAKTUR CAPITAL X -->
<!ENTITY yfr          "&#x1D536;" ><!--MATHEMATICAL FRAKTUR SMALL Y -->
<!ENTITY Yfr          "&#x1D51C;" ><!--MATHEMATICAL FRAKTUR CAPITAL Y -->
<!ENTITY zfr          "&#x1D537;" ><!--MATHEMATICAL FRAKTUR SMALL Z -->
<!ENTITY Zfr          "&#x02128;" ><!--BLACK-LETTER CAPITAL Z -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isomfrk.ent
No license file was found, but licenses were detected in source scan.

2000 W3C (MIT, INRIA, Keio),
All Rights Reserved.

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml10/xhtml1-transitional.dtd
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml10/xhtml1-strict.dtd
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Notations Module  ..................................... -->
<!-- file: xhtml-notations-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-notations-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//NOTATIONS XHTML Notations 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-notations-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Notations

   defines the following notations, many of these imported from
   other specifications and standards. When an existing FPI is
   known, it is incorporated here.
-->

<!-- XML Notations ..................................... -->
<!-- SGML and XML Notations ..................... -->

<!-- W3C XML 1.0 Recommendation -->
<!NOTATION w3c-xml
   PUBLIC "ISO 8879//NOTATION Extensible Markup Language (XML) 1.0//EN" >

<!-- XML 1.0 CDATA -->
<!NOTATION cdata
   PUBLIC "-//W3C//NOTATION XML 1.0: CDATA//EN" >

<!-- SGML Formal Public Identifiers -->
<!NOTATION fpi
   PUBLIC "ISO 8879:1986//NOTATION Formal Public Identifier//EN" >
```

```
<!-- XHTML Notations ..................................... -->

<!-- Length defined for cellpadding/cellspacing -->

<!-- nn for pixels or nn% for percentage length -->
<!NOTATION length
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Length//EN" >

<!-- space-separated list of link types -->
<!NOTATION linkTypes
  PUBLIC "-//W3C//NOTATION XHTML Datatype: LinkTypes//EN" >

<!-- single or comma-separated list of media descriptors -->
<!NOTATION mediaDesc
  PUBLIC "-//W3C//NOTATION XHTML Datatype: MediaDesc//EN" >

<!-- pixel, percentage, or relative -->
<!NOTATION multiLength
  PUBLIC "-//W3C//NOTATION XHTML Datatype: MultiLength//EN" >

<!-- one or more digits (NUMBER) -->
<!NOTATION number
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Number//EN" >

<!-- integer representing length in pixels -->
<!NOTATION pixels
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Pixels//EN" >

<!-- script expression -->
<!NOTATION script
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Script//EN" >

<!-- textual content -->
<!NOTATION text
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Text//EN" >

<!-- Imported Notations ................................. -->

<!-- a single character from [ISO10646] -->
<!NOTATION character
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Character//EN" >

<!-- a character encoding, as per [RFC2045] -->
<!NOTATION charset
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Charset//EN" >

<!-- a space separated list of character encodings, as per [RFC2045] -->
<!NOTATION charsets
```

```
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Charsets//EN" >

<!-- media type, as per [RFC2045] -->
<!NOTATION contentType
  PUBLIC "-//W3C//NOTATION XHTML Datatype: ContentType//EN" >

<!-- comma-separated list of media types, as per [RFC2045] -->
<!NOTATION contentTypes
  PUBLIC "-//W3C//NOTATION XHTML Datatype: ContentTypes//EN" >

<!-- date and time information. ISO date format -->
<!NOTATION datetime
  PUBLIC "-//W3C//NOTATION XHTML Datatype: Datetime//EN" >

<!-- a language code, as per [RFC3066] -->
<!NOTATION languageCode
  PUBLIC "-//W3C//NOTATION XHTML Datatype: LanguageCode//EN" >

<!-- a Uniform Resource Identifier, see [URI] -->
<!NOTATION uri
  PUBLIC "-//W3C//NOTATION XHTML Datatype: URI//EN" >

<!-- a space-separated list of Uniform Resource Identifiers, see [URI] -->
<!NOTATION uris
  PUBLIC "-//W3C//NOTATION XHTML Datatype: URIs//EN" >

<!-- end of xhtml-notations-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-notations-1.mod
No license file was found, but licenses were detected in source scan.

```
<!--
   File isoamso.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.
```

Entity names in this file are derived from files carrying the
following notice:

(C) International Organization for Standardization 1991
Permission to copy in any form is granted for use with
conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Added Math Symbols: Ordinary//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isoamso.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isoamso PUBLIC
"-//W3C//ENTITIES Added Math Symbols: Ordinary//EN//XML"
"http://www.w3.org/2003/entities/2007/isoamso.ent"
>
%isoamso;

-->

<!ENTITY ang "∠" ><!--ANGLE -->
<!ENTITY ange "⦤" ><!--ANGLE WITH UNDERBAR -->
<!ENTITY angmsd "∡" ><!--MEASURED ANGLE -->
<!ENTITY angmsdaa "⦨" ><!--MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING UP AND RIGHT -->
<!ENTITY angmsdab "⦩" ><!--MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING UP AND LEFT -->
<!ENTITY angmsdac "⦪" ><!--MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING DOWN AND RIGHT -->
<!ENTITY angmsdad "⦫" ><!--MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING DOWN AND LEFT -->
<!ENTITY angmsdae "⦬" ><!--MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING RIGHT AND UP -->
<!ENTITY angmsdaf "⦭" ><!--MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING LEFT AND UP -->
<!ENTITY angmsdag "⦮" ><!--MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING RIGHT AND DOWN -->
<!ENTITY angmsdah "⦯" ><!--MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW

POINTING LEFT AND DOWN -->
<!ENTITY angrtvb "⊾" ><!--RIGHT ANGLE WITH ARC -->
<!ENTITY angrtvbd "⦝" ><!--MEASURED RIGHT ANGLE WITH DOT -->
<!ENTITY bbrk "⎵" ><!--BOTTOM SQUARE BRACKET -->
<!ENTITY bbrktbrk "⎶" ><!--BOTTOM SQUARE BRACKET OVER TOP SQUARE BRACKET -
->
<!ENTITY bemptyv "⦰" ><!--REVERSED EMPTY SET -->
<!ENTITY beth "ℶ" ><!--BET SYMBOL -->
<!ENTITY boxbox "⧉" ><!--TWO JOINED SQUARES -->
<!ENTITY bprime "‵" ><!--REVERSED PRIME -->
<!ENTITY bsemi "⁏" ><!--REVERSED SEMICOLON -->
<!ENTITY cemptyv "⦲" ><!--EMPTY SET WITH SMALL CIRCLE ABOVE -->
<!ENTITY cirE "⧃" ><!--CIRCLE WITH TWO HORIZONTAL STROKES TO THE RIGHT -->
<!ENTITY cirscir "⧂" ><!--CIRCLE WITH SMALL CIRCLE TO THE RIGHT -->
<!ENTITY comp "∁" ><!--COMPLEMENT -->
<!ENTITY daleth "ℸ" ><!--DALET SYMBOL -->
<!ENTITY demptyv "⦱" ><!--EMPTY SET WITH OVERBAR -->
<!ENTITY ell "ℓ" ><!--SCRIPT SMALL L -->
<!ENTITY empty "∅" ><!--EMPTY SET -->
<!ENTITY emptyv "∅" ><!--EMPTY SET -->
<!ENTITY gimel "ℷ" ><!--GIMEL SYMBOL -->
<!ENTITY iiota "℩" ><!--TURNED GREEK SMALL LETTER IOTA -->
<!ENTITY image "ℑ" ><!--BLACK-LETTER CAPITAL I -->
<!ENTITY imath "ı" ><!--LATIN SMALL LETTER DOTLESS I -->
<!ENTITY jmath "ȷ" ><!--LATIN SMALL LETTER DOTLESS J -->
<!ENTITY laemptyv "⦴" ><!--EMPTY SET WITH LEFT ARROW ABOVE -->
<!ENTITY lltri "◺" ><!--LOWER LEFT TRIANGLE -->
<!ENTITY lrtri "⊿" ><!--RIGHT TRIANGLE -->
<!ENTITY mho "℧" ><!--INVERTED OHM SIGN -->
<!ENTITY nang "∠⃒" ><!--ANGLE with vertical line -->
<!ENTITY nexist "∄" ><!--THERE DOES NOT EXIST -->
<!ENTITY oS "Ⓢ" ><!--CIRCLED LATIN CAPITAL LETTER S -->
<!ENTITY planck "ℏ" ><!--PLANCK CONSTANT OVER TWO PI -->
<!ENTITY plankv "ℏ" ><!--PLANCK CONSTANT OVER TWO PI -->
<!ENTITY raemptyv "⦳" ><!--EMPTY SET WITH RIGHT ARROW ABOVE -->
<!ENTITY range "⦥" ><!--REVERSED ANGLE WITH UNDERBAR -->
<!ENTITY real "ℜ" ><!--BLACK-LETTER CAPITAL R -->
<!ENTITY tbrk "⎴" ><!--TOP SQUARE BRACKET -->
<!ENTITY trpezium "⏢" ><!--WHITE TRAPEZIUM -->
<!ENTITY ultri "◸" ><!--UPPER LEFT TRIANGLE -->
<!ENTITY urtri "◹" ><!--UPPER RIGHT TRIANGLE -->
<!ENTITY vzigzag "⦚" ><!--VERTICAL ZIGZAG LINE -->
<!ENTITY weierp "℘" ><!--SCRIPT CAPITAL P -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/mathml/mathml3/isoamso.ent

No license file was found, but licenses were detected in source scan.

```
<!--
   * This is an umbrella schema for the various schema documents used
   * to support XPath 3.0 functions and operators.
   *
   * The schema is made available under the terms of the W3C software notice and license
   * at http://www.w3.org/Consortium/Legal/copyright-software-19980720
   *
  -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/xpath-functions.xsd
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Basic Table Module  ..... -->
<!-- file: xhtml-basic-table-1.mod

   This is XHTML Basic, a proper subset of XHTML.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-basic-table-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Basic Tables 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-basic-table-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Basic Tables

      table, caption, tr, th, td

   This table module declares elements and attributes defining
   a table model based fundamentally on features found in the
   widely-deployed HTML 3.2 table model.  While this module
   mimics the content model and table attributes of HTML 3.2
   tables, the element types declared herein also includes all
   HTML 4 common and most of the HTML 4 table attributes.
-->

<!-- declare qualified element type names:
-->
<!ENTITY % table.qname  "table" >
```

```
<!ENTITY % caption.qname  "caption" >
<!ENTITY % tr.qname  "tr" >
<!ENTITY % th.qname  "th" >
<!ENTITY % td.qname  "td" >

<!-- horizontal alignment attributes for cell contents
-->
<!ENTITY % CellHAlign.attrib
   "align       ( left
              | center
              | right )              #IMPLIED"
>

<!-- vertical alignment attributes for cell contents
-->
<!ENTITY % CellVAlign.attrib
   "valign      ( top
              | middle
              | bottom )             #IMPLIED"
>

<!-- scope is simpler than axes attribute for common tables
-->
<!ENTITY % scope.attrib
   "scope       ( row | col  )         #IMPLIED"
>

<!-- table: Table Element ..... -->

<!ENTITY % table.element  "INCLUDE" >
<![%table.element;[
<!ENTITY % table.content
   "( %caption.qname;?, %tr.qname;+ )"
>
<!ELEMENT %table.qname;  %table.content; >
<!-- end of table.element -->]]>

<!ENTITY % table.attlist  "INCLUDE" >
<![%table.attlist;[
<!ATTLIST %table.qname;
    %Common.attrib;
    summary      %Text.datatype;         #IMPLIED
>
<!-- end of table.attlist -->]]>

<!-- caption: Table Caption ..... -->

<!ENTITY % caption.element  "INCLUDE" >
```

```
<![%caption.element;[
<!ENTITY % caption.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ELEMENT %caption.qname;  %caption.content; >
<!-- end of caption.element -->]]>

<!ENTITY % caption.attlist  "INCLUDE" >
<![%caption.attlist;[
<!ATTLIST %caption.qname;
    %Common.attrib;
>
<!-- end of caption.attlist -->]]>

<!-- tr: Table Row ..... -->

<!ENTITY % tr.element  "INCLUDE" >
<![%tr.element;[
<!ENTITY % tr.content  "( %th.qname; | %td.qname; )+" >
<!ELEMENT %tr.qname;  %tr.content; >
<!-- end of tr.element -->]]>

<!ENTITY % tr.attlist  "INCLUDE" >
<![%tr.attlist;[
<!ATTLIST %tr.qname;
    %Common.attrib;
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of tr.attlist -->]]>

<!-- th: Table Header Cell ..... -->

<!-- th is for header cells, td for data,
   but for cells acting as both use td
-->

<!ENTITY % th.element  "INCLUDE" >
<![%th.element;[
<!ENTITY % th.content
   "( #PCDATA | %FlowNoTable.mix; )*"
>
<!ELEMENT %th.qname;  %th.content; >
<!-- end of th.element -->]]>

<!ENTITY % th.attlist  "INCLUDE" >
<![%th.attlist;[
<!ATTLIST %th.qname;
```

```
    %Common.attrib;
    abbr        %Text.datatype;         #IMPLIED
    axis        CDATA                   #IMPLIED
    headers     IDREFS                  #IMPLIED
    %scope.attrib;
    rowspan     %Number.datatype;       '1'
    colspan     %Number.datatype;       '1'
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of th.attlist -->]]>

<!-- td: Table Data Cell ..... -->

<!ENTITY % td.element  "INCLUDE" >
<![%td.element;[
<!ENTITY % td.content
   "( #PCDATA | %FlowNoTable.mix; )*"
>
<!ELEMENT %td.qname;  %td.content; >
<!-- end of td.element -->]]>

<!ENTITY % td.attlist  "INCLUDE" >
<![%td.attlist;[
<!ATTLIST %td.qname;
    %Common.attrib;
    abbr        %Text.datatype;         #IMPLIED
    axis        CDATA                   #IMPLIED
    headers     IDREFS                  #IMPLIED
    %scope.attrib;
    rowspan     %Number.datatype;       '1'
    colspan     %Number.datatype;       '1'
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of td.attlist -->]]>

<!-- end of xhtml-basic-table-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-basic-table-1.mod
No license file was found, but licenses were detected in source scan.

<!--
   File isoamsn.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.
```

Copyright 1998 - 2009 W3C.

Use and distribution of this code are permitted under the terms of the
W3C Software Notice and License.
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

Please report any errors to David Carlisle
via the public W3C list www-math@w3.org.

The numeric character values assigned to each entity
(should) match the Unicode assignments in Unicode 5.x.

References to the VARIANT SELECTOR 1 character (︀)
should match the uses listed in Unicode Technical Report 25.

Entity names in this file are derived from files carrying the
following notice:

(C) International Organization for Standardization 1991
Permission to copy in any form is granted for use with
conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Added Math Symbols: Negated Relations//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isoamsn.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isoamsn PUBLIC
"-//W3C//ENTITIES Added Math Symbols: Negated Relations//EN//XML"
"http://www.w3.org/2003/entities/2007/isoamsn.ent"
>
%isoamsn;

-->

<!ENTITY gnap "⪊" ><!--GREATER-THAN AND NOT APPROXIMATE -->
<!ENTITY gne "⪈" ><!--GREATER-THAN AND SINGLE-LINE NOT EQUAL TO -->
<!ENTITY gnE "≩" ><!--GREATER-THAN BUT NOT EQUAL TO -->
<!ENTITY gnsim "⋧" ><!--GREATER-THAN BUT NOT EQUIVALENT TO -->

<!ENTITY gvnE "≩︀" ><!--GREATER-THAN BUT NOT EQUAL TO - with vertical stroke -->
<!ENTITY lnap "⪉" ><!--LESS-THAN AND NOT APPROXIMATE -->
<!ENTITY lne "⪇" ><!--LESS-THAN AND SINGLE-LINE NOT EQUAL TO -->
<!ENTITY lnE "≨" ><!--LESS-THAN BUT NOT EQUAL TO -->
<!ENTITY lnsim "⋦" ><!--LESS-THAN BUT NOT EQUIVALENT TO -->
<!ENTITY lvnE "≨︀" ><!--LESS-THAN BUT NOT EQUAL TO - with vertical stroke -->
<!ENTITY nap "≉" ><!--NOT ALMOST EQUAL TO -->
<!ENTITY napE "⩰̸" ><!--APPROXIMATELY EQUAL OR EQUAL TO with slash -->
<!ENTITY napid "≋̸" ><!--TRIPLE TILDE with slash -->
<!ENTITY ncong "≇" ><!--NEITHER APPROXIMATELY NOR ACTUALLY EQUAL TO -->
<!ENTITY ncongdot "⩭̸" ><!--CONGRUENT WITH DOT ABOVE with slash -->
<!ENTITY nequiv "≢" ><!--NOT IDENTICAL TO -->
<!ENTITY nge "≱" ><!--NEITHER GREATER-THAN NOR EQUAL TO -->
<!ENTITY ngE "≧̸" ><!--GREATER-THAN OVER EQUAL TO with slash -->
<!ENTITY nges "⩾̸" ><!--GREATER-THAN OR SLANTED EQUAL TO with slash -->
<!ENTITY nGg "⋙̸" ><!--VERY MUCH GREATER-THAN with slash -->
<!ENTITY ngsim "≵" ><!--NEITHER GREATER-THAN NOR EQUIVALENT TO -->
<!ENTITY ngt "≯" ><!--NOT GREATER-THAN -->
<!ENTITY nGt "≫⃒" ><!--MUCH GREATER THAN with vertical line -->
<!ENTITY nGtv "≫̸" ><!--MUCH GREATER THAN with slash -->
<!ENTITY nle "≰" ><!--NEITHER LESS-THAN NOR EQUAL TO -->
<!ENTITY nlE "≦̸" ><!--LESS-THAN OVER EQUAL TO with slash -->
<!ENTITY nles "⩽̸" ><!--LESS-THAN OR SLANTED EQUAL TO with slash -->
<!ENTITY nLl "⋘̸" ><!--VERY MUCH LESS-THAN with slash -->
<!ENTITY nlsim "≴" ><!--NEITHER LESS-THAN NOR EQUIVALENT TO -->
<!ENTITY nlt "≮" ><!--NOT LESS-THAN -->
<!ENTITY nLt "≪⃒" ><!--MUCH LESS THAN with vertical line -->
<!ENTITY nltri "⋪" ><!--NOT NORMAL SUBGROUP OF -->
<!ENTITY nltrie "⋬" ><!--NOT NORMAL SUBGROUP OF OR EQUAL TO -->
<!ENTITY nLtv "≪̸" ><!--MUCH LESS THAN with slash -->
<!ENTITY nmid "∤" ><!--DOES NOT DIVIDE -->
<!ENTITY npar "∦" ><!--NOT PARALLEL TO -->
<!ENTITY npr "⊀" ><!--DOES NOT PRECEDE -->
<!ENTITY nprcue "⋠" ><!--DOES NOT PRECEDE OR EQUAL -->
<!ENTITY npre "⪯̸" ><!--PRECEDES ABOVE SINGLE-LINE EQUALS SIGN with slash -->
<!ENTITY nrtri "⋫" ><!--DOES NOT CONTAIN AS NORMAL SUBGROUP -->
<!ENTITY nrtrie "⋭" ><!--DOES NOT CONTAIN AS NORMAL SUBGROUP OR EQUAL -->
<!ENTITY nsc "⊁" ><!--DOES NOT SUCCEED -->
<!ENTITY nsccue "⋡" ><!--DOES NOT SUCCEED OR EQUAL -->
<!ENTITY nsce "⪰̸" ><!--SUCCEEDS ABOVE SINGLE-LINE EQUALS SIGN with slash -->
<!ENTITY nsim "≁" ><!--NOT TILDE -->
<!ENTITY nsime "≄" ><!--NOT ASYMPTOTICALLY EQUAL TO -->

<!ENTITY nsmid "∤" ><!--DOES NOT DIVIDE -->
<!ENTITY nspar "∦" ><!--NOT PARALLEL TO -->
<!ENTITY nsqsube "⋢" ><!--NOT SQUARE IMAGE OF OR EQUAL TO -->
<!ENTITY nsqsupe "⋣" ><!--NOT SQUARE ORIGINAL OF OR EQUAL TO -->
<!ENTITY nsub "⊄" ><!--NOT A SUBSET OF -->
<!ENTITY nsube "⊈" ><!--NEITHER A SUBSET OF NOR EQUAL TO -->
<!ENTITY nsubE "⫅̸" ><!--SUBSET OF ABOVE EQUALS SIGN with slash -->
<!ENTITY nsup "⊅" ><!--NOT A SUPERSET OF -->
<!ENTITY nsupe "⊉" ><!--NEITHER A SUPERSET OF NOR EQUAL TO -->
<!ENTITY nsupE "⫆̸" ><!--SUPERSET OF ABOVE EQUALS SIGN with slash -->
<!ENTITY ntgl "≹" ><!--NEITHER GREATER-THAN NOR LESS-THAN -->
<!ENTITY ntlg "≸" ><!--NEITHER LESS-THAN NOR GREATER-THAN -->
<!ENTITY nvap "≍⃒" ><!--EQUIVALENT TO with vertical line -->
<!ENTITY nvdash "⊬" ><!--DOES NOT PROVE -->
<!ENTITY nvDash "⊭" ><!--NOT TRUE -->
<!ENTITY nVdash "⊮" ><!--DOES NOT FORCE -->
<!ENTITY nVDash "⊯" ><!--NEGATED DOUBLE VERTICAL BAR DOUBLE RIGHT TURNSTILE -->
<!ENTITY nvge "≥⃒" ><!--GREATER-THAN OR EQUAL TO with vertical line -->
<!ENTITY nvgt ">⃒" ><!--GREATER-THAN SIGN with vertical line -->
<!ENTITY nvle "≤⃒" ><!--LESS-THAN OR EQUAL TO with vertical line -->
<!ENTITY nvlt "<⃒" ><!--LESS-THAN SIGN with vertical line -->
<!ENTITY nvltrie "⊴⃒" ><!--NORMAL SUBGROUP OF OR EQUAL TO with vertical line -->
<!ENTITY nvrtrie "⊵⃒" ><!--CONTAINS AS NORMAL SUBGROUP OR EQUAL TO with vertical line -->
<!ENTITY nvsim "∼⃒" ><!--TILDE OPERATOR with vertical line -->
<!ENTITY parsim "⫳" ><!--PARALLEL WITH TILDE OPERATOR -->
<!ENTITY prnap "⪹" ><!--PRECEDES ABOVE NOT ALMOST EQUAL TO -->
<!ENTITY prnE "⪵" ><!--PRECEDES ABOVE NOT EQUAL TO -->
<!ENTITY prnsim "⋨" ><!--PRECEDES BUT NOT EQUIVALENT TO -->
<!ENTITY rnmid "⫮" ><!--DOES NOT DIVIDE WITH REVERSED NEGATION SLASH -->
<!ENTITY scnap "⪺" ><!--SUCCEEDS ABOVE NOT ALMOST EQUAL TO -->
<!ENTITY scnE "⪶" ><!--SUCCEEDS ABOVE NOT EQUAL TO -->
<!ENTITY scnsim "⋩" ><!--SUCCEEDS BUT NOT EQUIVALENT TO -->
<!ENTITY simne "≆" ><!--APPROXIMATELY BUT NOT ACTUALLY EQUAL TO -->
<!ENTITY solbar "⌿" ><!--APL FUNCTIONAL SYMBOL SLASH BAR -->
<!ENTITY subne "⊊" ><!--SUBSET OF WITH NOT EQUAL TO -->
<!ENTITY subnE "⫋" ><!--SUBSET OF ABOVE NOT EQUAL TO -->
<!ENTITY supne "⊋" ><!--SUPERSET OF WITH NOT EQUAL TO -->
<!ENTITY supnE "⫌" ><!--SUPERSET OF ABOVE NOT EQUAL TO -->
<!ENTITY vnsub "⊂⃒" ><!--SUBSET OF with vertical line -->
<!ENTITY vnsup "⊃⃒" ><!--SUPERSET OF with vertical line -->
<!ENTITY vsubne "⊊︀" ><!--SUBSET OF WITH NOT EQUAL TO - variant with stroke through bottom members -->
<!ENTITY vsubnE "⫋︀" ><!--SUBSET OF ABOVE NOT EQUAL TO - variant with stroke through bottom members -->
<!ENTITY vsupne "⊋︀" ><!--SUPERSET OF WITH NOT EQUAL TO - variant with

stroke through bottom members -->
<!ENTITY vsupnE "⫌︀" ><!--SUPERSET OF ABOVE NOT EQUAL TO - variant with stroke through bottom members -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isoamsn.ent
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Graphics Attribute Module ..... -->
<!-- file: svg-graphics-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-graphics-attrib.mod,v 1.2 2002/04/20 18:07:43 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Graphics Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-graphics-attrib.mod"

   ..... -->

<!-- Graphics Attribute

     display, image-rendering, pointer-events, shape-rendering,
     text-rendering, visibility

   This module defines the Graphics attribute set.
-->

<!ENTITY % SVG.display.attrib
  "display (inline | block | list-item | run-in | compact | marker |
          table | inline-table | table-row-group | table-header-group |
          table-footer-group | table-row | table-column-group |
          table-column | table-cell | table-caption | none | inherit)
          #IMPLIED"
>

<!ENTITY % SVG.image-rendering.attrib
  "image-rendering (auto | optimizeSpeed | optimizeQuality | inherit)
                #IMPLIED"
>

<!ENTITY % SVG.pointer-events.attrib
  "pointer-events (visiblePainted | visibleFill | visibleStroke | visible |
                painted | fill | stroke | all | none | inherit)

```
                #IMPLIED"
>

<!ENTITY % SVG.shape-rendering.attrib
  "shape-rendering ( auto | optimizeSpeed | crispEdges | geometricPrecision |
                inherit ) #IMPLIED"
>

<!ENTITY % SVG.text-rendering.attrib
  "text-rendering ( auto | optimizeSpeed | optimizeLegibility |
                geometricPrecision | inherit ) #IMPLIED"
>

<!ENTITY % SVG.visibility.attrib
  "visibility ( visible | hidden | inherit ) #IMPLIED"
>

<!ENTITY % SVG.Graphics.extra.attrib "" >

<!ENTITY % SVG.Graphics.attrib
  "%SVG.display.attrib;
   %SVG.image-rendering.attrib;
   %SVG.pointer-events.attrib;
   %SVG.shape-rendering.attrib;
   %SVG.text-rendering.attrib;
   %SVG.visibility.attrib;
   %SVG.Graphics.extra.attrib;"
>

<!-- end of svg-graphics-attrib.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-graphics-attrib.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Datatypes Module  ..................................... -->
<!-- file: xhtml-datatypes-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-datatypes-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Datatypes 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-datatypes-1.mod"
```

```
    Revisions:
    (none)
    ..... -->

<!-- Datatypes

    defines containers for the following datatypes, many of
    these imported from other specifications and standards.
-->

<!-- Length defined for cellpadding/cellspacing -->

<!-- nn for pixels or nn% for percentage length -->
<!ENTITY % Length.datatype "CDATA" >

<!-- space-separated list of link types -->
<!ENTITY % LinkTypes.datatype "NMTOKENS" >

<!-- single or comma-separated list of media descriptors -->
<!ENTITY % MediaDesc.datatype "CDATA" >

<!-- pixel, percentage, or relative -->
<!ENTITY % MultiLength.datatype "CDATA" >

<!-- one or more digits (NUMBER) -->
<!ENTITY % Number.datatype "CDATA" >

<!-- integer representing length in pixels -->
<!ENTITY % Pixels.datatype "CDATA" >

<!-- script expression -->
<!ENTITY % Script.datatype "CDATA" >

<!-- textual content -->
<!ENTITY % Text.datatype "CDATA" >

<!-- Imported Datatypes ..... -->

<!-- a single character from [ISO10646] -->
<!ENTITY % Character.datatype "CDATA" >

<!-- a character encoding, as per [RFC2045] -->
<!ENTITY % Charset.datatype "CDATA" >

<!-- a space separated list of character encodings, as per [RFC2045] -->
<!ENTITY % Charsets.datatype "CDATA" >
```

```
<!-- media type, as per [RFC2045] -->
<!ENTITY % ContentType.datatype "CDATA" >

<!-- comma-separated list of media types, as per [RFC2045] -->
<!ENTITY % ContentTypes.datatype "CDATA" >

<!-- date and time information. ISO date format -->
<!ENTITY % Datetime.datatype "CDATA" >

<!-- formal public identifier, as per [ISO8879] -->
<!ENTITY % FPI.datatype "CDATA" >

<!-- a language code, as per [RFC1766] -->
<!ENTITY % LanguageCode.datatype "NMTOKEN" >

<!-- a Uniform Resource Identifier, see [URI] -->
<!ENTITY % URI.datatype "CDATA" >

<!-- a space-separated list of Uniform Resource Identifiers, see [URI] -->
<!ENTITY % URIs.datatype "CDATA" >

<!-- end of xhtml-datatypes-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-datatypes-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- SVG 1.1 Container Attribute Module ..................................... -->
<!-- file: svg-container-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-container-attrib.mod,v 1.2 2002/04/20 18:07:42 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Container Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-container-attrib.mod"

   ....................................................................... -->

<!-- Container Attribute

     enable-background

   This module defines the Container attribute set.
```

```
-->

<!-- 'enable-background' property/attribute value (e.g., 'new', 'accumulate') -->
<!ENTITY % EnableBackgroundValue.datatype "CDATA" >

<!ENTITY % SVG.enable-background.attrib
  "enable-background %EnableBackgroundValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.Container.extra.attrib "" >

<!ENTITY % SVG.Container.attrib
  "%SVG.enable-background.attrib;
   %SVG.Container.extra.attrib;"
>

<!-- end of svg-container-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-container-attrib.mod
No license file was found, but licenses were detected in source scan.

<!--
   File isoamsb.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

   References to the VARIANT SELECTOR 1 character (&#x0FE00;)
   should match the uses listed in Unicode Technical Report 25.

   Entity names in this file are derived from files carrying the
   following notice:

   (C) International Organization for Standardization 1991
   Permission to copy in any form is granted for use with
   conforming SGML systems and applications as defined in
```

ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Added Math Symbols: Binary Operators//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isoamsb.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isoamsb PUBLIC
"-//W3C//ENTITIES Added Math Symbols: Binary Operators//EN//XML"
"http://www.w3.org/2003/entities/2007/isoamsb.ent"
>
%isoamsb;

-->

<!ENTITY ac "∾" ><!--INVERTED LAZY S -->
<!ENTITY acE "∾̳" ><!--INVERTED LAZY S with double underline -->
<!ENTITY amalg "⨿" ><!--AMALGAMATION OR COPRODUCT -->
<!ENTITY barvee "⊽" ><!--NOR -->
<!ENTITY barwed "⌅" ><!--PROJECTIVE -->
<!ENTITY Barwed "⌆" ><!--PERSPECTIVE -->
<!ENTITY bsolb "⧅" ><!--SQUARED FALLING DIAGONAL SLASH -->
<!ENTITY Cap "⋒" ><!--DOUBLE INTERSECTION -->
<!ENTITY capand "⩄" ><!--INTERSECTION WITH LOGICAL AND -->
<!ENTITY capbrcup "⩉" ><!--INTERSECTION ABOVE BAR ABOVE UNION -->
<!ENTITY capcap "⩋" ><!--INTERSECTION BESIDE AND JOINED WITH INTERSECTION --
>
<!ENTITY capcup "⩇" ><!--INTERSECTION ABOVE UNION -->
<!ENTITY capdot "⩀" ><!--INTERSECTION WITH DOT -->
<!ENTITY caps "∩︀" ><!--INTERSECTION with serifs -->
<!ENTITY ccaps "⩍" ><!--CLOSED INTERSECTION WITH SERIFS -->
<!ENTITY ccups "⩌" ><!--CLOSED UNION WITH SERIFS -->
<!ENTITY ccupssm "⩐" ><!--CLOSED UNION WITH SERIFS AND SMASH PRODUCT -->
<!ENTITY coprod "∐" ><!--N-ARY COPRODUCT -->
<!ENTITY Cup "⋓" ><!--DOUBLE UNION -->
<!ENTITY cupbrcap "⩈" ><!--UNION ABOVE BAR ABOVE INTERSECTION -->
<!ENTITY cupcap "⩆" ><!--UNION ABOVE INTERSECTION -->
<!ENTITY cupcup "⩊" ><!--UNION BESIDE AND JOINED WITH UNION -->
<!ENTITY cupdot "⊍" ><!--MULTISET MULTIPLICATION -->
<!ENTITY cupor "⩅" ><!--UNION WITH LOGICAL OR -->

<!ENTITY cups "∪︀" ><!--UNION with serifs -->
<!ENTITY cuvee "⋎" ><!--CURLY LOGICAL OR -->
<!ENTITY cuwed "⋏" ><!--CURLY LOGICAL AND -->
<!ENTITY dagger "†" ><!--DAGGER -->
<!ENTITY Dagger "‡" ><!--DOUBLE DAGGER -->
<!ENTITY diam "⋄" ><!--DIAMOND OPERATOR -->
<!ENTITY divonx "⋇" ><!--DIVISION TIMES -->
<!ENTITY eplus "⩱" ><!--EQUALS SIGN ABOVE PLUS SIGN -->
<!ENTITY hercon "⊹" ><!--HERMITIAN CONJUGATE MATRIX -->
<!ENTITY intcal "⊺" ><!--INTERCALATE -->
<!ENTITY iprod "⨼" ><!--INTERIOR PRODUCT -->
<!ENTITY loplus "⨭" ><!--PLUS SIGN IN LEFT HALF CIRCLE -->
<!ENTITY lotimes "⨴" ><!--MULTIPLICATION SIGN IN LEFT HALF CIRCLE -->
<!ENTITY lthree "⋋" ><!--LEFT SEMIDIRECT PRODUCT -->
<!ENTITY ltimes "⋉" ><!--LEFT NORMAL FACTOR SEMIDIRECT PRODUCT -->
<!ENTITY midast "*" ><!--ASTERISK -->
<!ENTITY minusb "⊟" ><!--SQUARED MINUS -->
<!ENTITY minusd "∸" ><!--DOT MINUS -->
<!ENTITY minusdu "⨪" ><!--MINUS SIGN WITH DOT BELOW -->
<!ENTITY ncap "⩃" ><!--INTERSECTION WITH OVERBAR -->
<!ENTITY ncup "⩂" ><!--UNION WITH OVERBAR -->
<!ENTITY oast "⊛" ><!--CIRCLED ASTERISK OPERATOR -->
<!ENTITY ocir "⊚" ><!--CIRCLED RING OPERATOR -->
<!ENTITY odash "⊝" ><!--CIRCLED DASH -->
<!ENTITY odiv "⨸" ><!--CIRCLED DIVISION SIGN -->
<!ENTITY odot "⊙" ><!--CIRCLED DOT OPERATOR -->
<!ENTITY odsold "⦼" ><!--CIRCLED ANTICLOCKWISE-ROTATED DIVISION SIGN -->
<!ENTITY ofcir "⦿" ><!--CIRCLED BULLET -->
<!ENTITY ogt "⧁" ><!--CIRCLED GREATER-THAN -->
<!ENTITY ohbar "⦵" ><!--CIRCLE WITH HORIZONTAL BAR -->
<!ENTITY olcir "⦾" ><!--CIRCLED WHITE BULLET -->
<!ENTITY olt "⧀" ><!--CIRCLED LESS-THAN -->
<!ENTITY omid "⦶" ><!--CIRCLED VERTICAL BAR -->
<!ENTITY ominus "⊖" ><!--CIRCLED MINUS -->
<!ENTITY opar "⦷" ><!--CIRCLED PARALLEL -->
<!ENTITY operp "⦹" ><!--CIRCLED PERPENDICULAR -->
<!ENTITY oplus "⊕" ><!--CIRCLED PLUS -->
<!ENTITY osol "⊘" ><!--CIRCLED DIVISION SLASH -->
<!ENTITY otimes "⊗" ><!--CIRCLED TIMES -->
<!ENTITY Otimes "⨷" ><!--MULTIPLICATION SIGN IN DOUBLE CIRCLE -->
<!ENTITY otimesas "⨶" ><!--CIRCLED MULTIPLICATION SIGN WITH CIRCUMFLEX ACCENT -->
<!ENTITY ovbar "⌽" ><!--APL FUNCTIONAL SYMBOL CIRCLE STILE -->
<!ENTITY plusacir "⨣" ><!--PLUS SIGN WITH CIRCUMFLEX ACCENT ABOVE -->
<!ENTITY plusb "⊞" ><!--SQUARED PLUS -->
<!ENTITY pluscir "⨢" ><!--PLUS SIGN WITH SMALL CIRCLE ABOVE -->
<!ENTITY plusdo "∔" ><!--DOT PLUS -->
<!ENTITY plusdu "⨥" ><!--PLUS SIGN WITH DOT BELOW -->

```
<!ENTITY pluse         "&#x02A72;" ><!--PLUS SIGN ABOVE EQUALS SIGN -->
<!ENTITY plussim       "&#x02A26;" ><!--PLUS SIGN WITH TILDE BELOW -->
<!ENTITY plustwo       "&#x02A27;" ><!--PLUS SIGN WITH SUBSCRIPT TWO -->
<!ENTITY prod          "&#x0220F;" ><!--N-ARY PRODUCT -->
<!ENTITY race          "&#x0223D;&#x00331;" ><!--REVERSED TILDE with underline -->
<!ENTITY roplus        "&#x02A2E;" ><!--PLUS SIGN IN RIGHT HALF CIRCLE -->
<!ENTITY rotimes       "&#x02A35;" ><!--MULTIPLICATION SIGN IN RIGHT HALF CIRCLE -->
<!ENTITY rthree        "&#x022CC;" ><!--RIGHT SEMIDIRECT PRODUCT -->
<!ENTITY rtimes        "&#x022CA;" ><!--RIGHT NORMAL FACTOR SEMIDIRECT PRODUCT -->
<!ENTITY sdot          "&#x022C5;" ><!--DOT OPERATOR -->
<!ENTITY sdotb         "&#x022A1;" ><!--SQUARED DOT OPERATOR -->
<!ENTITY setmn         "&#x02216;" ><!--SET MINUS -->
<!ENTITY simplus       "&#x02A24;" ><!--PLUS SIGN WITH TILDE ABOVE -->
<!ENTITY smashp        "&#x02A33;" ><!--SMASH PRODUCT -->
<!ENTITY solb          "&#x029C4;" ><!--SQUARED RISING DIAGONAL SLASH -->
<!ENTITY sqcap         "&#x02293;" ><!--SQUARE CAP -->
<!ENTITY sqcaps        "&#x02293;&#x0FE00;" ><!--SQUARE CAP with serifs -->
<!ENTITY sqcup         "&#x02294;" ><!--SQUARE CUP -->
<!ENTITY sqcups        "&#x02294;&#x0FE00;" ><!--SQUARE CUP with serifs -->
<!ENTITY ssetmn        "&#x02216;" ><!--SET MINUS -->
<!ENTITY sstarf        "&#x022C6;" ><!--STAR OPERATOR -->
<!ENTITY subdot        "&#x02ABD;" ><!--SUBSET WITH DOT -->
<!ENTITY sum           "&#x02211;" ><!--N-ARY SUMMATION -->
<!ENTITY supdot        "&#x02ABE;" ><!--SUPERSET WITH DOT -->
<!ENTITY timesb        "&#x022A0;" ><!--SQUARED TIMES -->
<!ENTITY timesbar      "&#x02A31;" ><!--MULTIPLICATION SIGN WITH UNDERBAR -->
<!ENTITY timesd        "&#x02A30;" ><!--MULTIPLICATION SIGN WITH DOT ABOVE -->
<!ENTITY tridot        "&#x025EC;" ><!--WHITE UP-POINTING TRIANGLE WITH DOT -->
<!ENTITY triminus      "&#x02A3A;" ><!--MINUS SIGN IN TRIANGLE -->
<!ENTITY triplus       "&#x02A39;" ><!--PLUS SIGN IN TRIANGLE -->
<!ENTITY trisb         "&#x029CD;" ><!--TRIANGLE WITH SERIFS AT BOTTOM -->
<!ENTITY tritime       "&#x02A3B;" ><!--MULTIPLICATION SIGN IN TRIANGLE -->
<!ENTITY uplus         "&#x0228E;" ><!--MULTISET UNION -->
<!ENTITY veebar        "&#x022BB;" ><!--XOR -->
<!ENTITY wedbar        "&#x02A5F;" ><!--LOGICAL AND WITH UNDERBAR -->
<!ENTITY wreath        "&#x02240;" ><!--WREATH PRODUCT -->
<!ENTITY xcap          "&#x022C2;" ><!--N-ARY INTERSECTION -->
<!ENTITY xcirc         "&#x025EF;" ><!--LARGE CIRCLE -->
<!ENTITY xcup          "&#x022C3;" ><!--N-ARY UNION -->
<!ENTITY xdtri         "&#x025BD;" ><!--WHITE DOWN-POINTING TRIANGLE -->
<!ENTITY xodot         "&#x02A00;" ><!--N-ARY CIRCLED DOT OPERATOR -->
<!ENTITY xoplus        "&#x02A01;" ><!--N-ARY CIRCLED PLUS OPERATOR -->
<!ENTITY xotime        "&#x02A02;" ><!--N-ARY CIRCLED TIMES OPERATOR -->
<!ENTITY xsqcup        "&#x02A06;" ><!--N-ARY SQUARE UNION OPERATOR -->
<!ENTITY xuplus        "&#x02A04;" ><!--N-ARY UNION OPERATOR WITH PLUS -->
<!ENTITY xutri         "&#x025B3;" ><!--WHITE UP-POINTING TRIANGLE -->
<!ENTITY xvee          "&#x022C1;" ><!--N-ARY LOGICAL OR -->
<!ENTITY xwedge        "&#x022C0;" ><!--N-ARY LOGICAL AND -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isoamsb.ent
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Qname Module  ................................................... -->
<!-- file: xhtml-qname-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-qname-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Qualified Names 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-qname-1.mod"

   Revisions:
   #2000-10-22: added qname declarations for ruby elements
   ....................................................................... -->

<!-- XHTML Qname (Qualified Name) Module

   This module is contained in two parts, labeled Section 'A' and 'B':

     Section A declares parameter entities to support namespace-
     qualified names, namespace declarations, and name prefixing
     for XHTML and extensions.

     Section B declares parameter entities used to provide
     namespace-qualified names for all XHTML element types:

       %applet.qname;   the xmlns-qualified name for <applet>
       %base.qname;     the xmlns-qualified name for <base>
       ...

   XHTML extensions would create a module similar to this one.
   Included in the XHTML distribution is a template module
   ('template-qname-1.mod') suitable for this purpose.
-->

<!-- Section A: XHTML XML Namespace Framework :::::::::::::::::::: -->

<!-- 1. Declare a %XHTML.prefixed; conditional section keyword, used
      to activate namespace prefixing. The default value should
      inherit '%NS.prefixed;' from the DTD driver, so that unless
```

```
      overridden, the default behaviour follows the overall DTD
      prefixing scheme.
-->
<!ENTITY % NS.prefixed "IGNORE" >
<!ENTITY % XHTML.prefixed "%NS.prefixed;" >

<!-- By default, we always permit XHTML attribute collections to have
   namespace-qualified prefixes as well.
-->
<!ENTITY % XHTML.global.attrs.prefixed "INCLUDE" >
<!-- By default, we allow the XML Schema attributes on the root
   element.
-->
<!ENTITY % XHTML.xsi.attrs "INCLUDE" >

<!-- 2. Declare a parameter entity (eg., %XHTML.xmlns;) containing
      the URI reference used to identify the XHTML namespace:
-->
<!ENTITY % XHTML.xmlns  "http://www.w3.org/1999/xhtml" >

<!-- 3. Declare parameter entities (eg., %XHTML.prefix;) containing
      the default namespace prefix string(s) to use when prefixing
      is enabled. This may be overridden in the DTD driver or the
      internal subset of an document instance. If no default prefix
      is desired, this may be declared as an empty string.

   NOTE: As specified in [XMLNAMES], the namespace prefix serves
   as a proxy for the URI reference, and is not in itself significant.
-->
<!ENTITY % XHTML.prefix  "xhtml" >

<!-- 4. Declare parameter entities (eg., %XHTML.pfx;) containing the
      colonized prefix(es) (eg., '%XHTML.prefix;:') used when
      prefixing is active, an empty string when it is not.
-->
<![%XHTML.prefixed;[
<!ENTITY % XHTML.pfx  "%XHTML.prefix;:" >
]]>
<!ENTITY % XHTML.pfx  "" >

<!-- declare qualified name extensions here ..... -->
<!ENTITY % xhtml-qname-extra.mod "" >
%xhtml-qname-extra.mod;

<!-- 5. The parameter entity %XHTML.xmlns.extra.attrib; may be
      redeclared to contain any non-XHTML namespace declaration
      attributes for namespaces embedded in XHTML. The default
      is an empty string.  XLink should be included here if used
```

```
      in the DTD.
-->
<!ENTITY % XHTML.xmlns.extra.attrib "" >

<!-- The remainder of Section A is only followed in XHTML, not extensions. -->

<!-- Declare a parameter entity %NS.decl.attrib; containing
   all XML Namespace declarations used in the DTD, plus the
   xmlns declaration for XHTML, its form dependent on whether
   prefixing is active.
-->
<!ENTITY % XHTML.xmlns.attrib.prefixed
   "xmlns:%XHTML.prefix;  %URI.datatype;   #FIXED '%XHTML.xmlns;'"
>
<![%XHTML.prefixed;[
<!ENTITY % NS.decl.attrib
   "%XHTML.xmlns.attrib.prefixed;
    %XHTML.xmlns.extra.attrib;"
>
]]>
<!ENTITY % NS.decl.attrib
   "%XHTML.xmlns.extra.attrib;"
>

<!-- Declare a parameter entity %XSI.prefix as a prefix to use for XML
   Schema Instance attributes.
-->
<!ENTITY % XSI.prefix "xsi" >

<!ENTITY % XSI.xmlns "http://www.w3.org/2001/XMLSchema-instance" >

<!-- Declare a parameter entity %XSI.xmlns.attrib as support for the
   schemaLocation attribute, since this is legal throughout the DTD.
-->
<!ENTITY % XSI.xmlns.attrib
   "xmlns:%XSI.prefix;  %URI.datatype;   #FIXED '%XSI.xmlns;'" >

<!-- This is a placeholder for future XLink support.
-->
<!ENTITY % XLINK.xmlns.attrib "" >

<!-- This is the attribute for the XML Schema namespace - XHTML
   Modularization is also expressed in XML Schema, and it needs to
  be legal to declare the XML Schema namespace and the
  schemaLocation attribute on the root element of XHTML family
  documents.
-->
<![%XHTML.xsi.attrs;[
```

```
<!ENTITY % XSI.prefix "xsi" >
<!ENTITY % XSI.pfx "%XSI.prefix;:" >
<!ENTITY % XSI.xmlns "http://www.w3.org/2001/XMLSchema-instance" >

<!ENTITY % XSI.xmlns.attrib
   "xmlns:%XSI.prefix;  %URI.datatype;    #FIXED '%XSI.xmlns;'"
>
]]>
<!ENTITY % XSI.prefix "" >
<!ENTITY % XSI.pfx "" >
<!ENTITY % XSI.xmlns.attrib "" >


<!-- Declare a parameter entity %NS.decl.attrib; containing all
   XML namespace declaration attributes used by XHTML, including
   a default xmlns attribute when prefixing is inactive.
-->
<![%XHTML.prefixed;[
<!ENTITY % XHTML.xmlns.attrib
   "%NS.decl.attrib;
    %XSI.xmlns.attrib;
    %XLINK.xmlns.attrib;"
>
]]>
<!ENTITY % XHTML.xmlns.attrib
   "xmlns       %URI.datatype;          #FIXED '%XHTML.xmlns;'
    %NS.decl.attrib;
    %XSI.xmlns.attrib;
    %XLINK.xmlns.attrib;"
>

<!-- placeholder for qualified name redeclarations -->
<!ENTITY % xhtml-qname.redecl "" >
%xhtml-qname.redecl;

<!-- Section B: XHTML Qualified Names :::::::::::::::::::::::::::::: -->

<!-- 6. This section declares parameter entities used to provide
      namespace-qualified names for all XHTML element types.
-->

<!-- module:  xhtml-applet-1.mod -->
<!ENTITY % applet.qname  "%XHTML.pfx;applet" >

<!-- module:  xhtml-base-1.mod -->
<!ENTITY % base.qname    "%XHTML.pfx;base" >

<!-- module:  xhtml-bdo-1.mod -->
```

```
<!ENTITY % bdo.qname     "%XHTML.pfx;bdo" >

<!-- module:  xhtml-blkphras-1.mod -->
<!ENTITY % address.qname "%XHTML.pfx;address" >
<!ENTITY % blockquote.qname  "%XHTML.pfx;blockquote" >
<!ENTITY % pre.qname     "%XHTML.pfx;pre" >
<!ENTITY % h1.qname      "%XHTML.pfx;h1" >
<!ENTITY % h2.qname      "%XHTML.pfx;h2" >
<!ENTITY % h3.qname      "%XHTML.pfx;h3" >
<!ENTITY % h4.qname      "%XHTML.pfx;h4" >
<!ENTITY % h5.qname      "%XHTML.pfx;h5" >
<!ENTITY % h6.qname      "%XHTML.pfx;h6" >

<!-- module:  xhtml-blkpres-1.mod -->
<!ENTITY % hr.qname      "%XHTML.pfx;hr" >

<!-- module:  xhtml-blkstruct-1.mod -->
<!ENTITY % div.qname     "%XHTML.pfx;div" >
<!ENTITY % p.qname       "%XHTML.pfx;p" >

<!-- module:  xhtml-edit-1.mod -->
<!ENTITY % ins.qname     "%XHTML.pfx;ins" >
<!ENTITY % del.qname     "%XHTML.pfx;del" >

<!-- module:  xhtml-form-1.mod -->
<!ENTITY % form.qname    "%XHTML.pfx;form" >
<!ENTITY % label.qname   "%XHTML.pfx;label" >
<!ENTITY % input.qname   "%XHTML.pfx;input" >
<!ENTITY % select.qname  "%XHTML.pfx;select" >
<!ENTITY % optgroup.qname  "%XHTML.pfx;optgroup" >
<!ENTITY % option.qname  "%XHTML.pfx;option" >
<!ENTITY % textarea.qname  "%XHTML.pfx;textarea" >
<!ENTITY % fieldset.qname  "%XHTML.pfx;fieldset" >
<!ENTITY % legend.qname  "%XHTML.pfx;legend" >
<!ENTITY % button.qname  "%XHTML.pfx;button" >

<!-- module:  xhtml-hypertext-1.mod -->
<!ENTITY % a.qname       "%XHTML.pfx;a" >

<!-- module:  xhtml-image-1.mod -->
<!ENTITY % img.qname     "%XHTML.pfx;img" >

<!-- module:  xhtml-inlphras-1.mod -->
<!ENTITY % abbr.qname    "%XHTML.pfx;abbr" >
<!ENTITY % acronym.qname "%XHTML.pfx;acronym" >
<!ENTITY % cite.qname    "%XHTML.pfx;cite" >
<!ENTITY % code.qname    "%XHTML.pfx;code" >
<!ENTITY % dfn.qname     "%XHTML.pfx;dfn" >
```

```
<!ENTITY % em.qname      "%XHTML.pfx;em" >
<!ENTITY % kbd.qname     "%XHTML.pfx;kbd" >
<!ENTITY % q.qname       "%XHTML.pfx;q" >
<!ENTITY % samp.qname    "%XHTML.pfx;samp" >
<!ENTITY % strong.qname  "%XHTML.pfx;strong" >
<!ENTITY % var.qname     "%XHTML.pfx;var" >

<!-- module:  xhtml-inlpres-1.mod -->
<!ENTITY % b.qname       "%XHTML.pfx;b" >
<!ENTITY % big.qname     "%XHTML.pfx;big" >
<!ENTITY % i.qname       "%XHTML.pfx;i" >
<!ENTITY % small.qname   "%XHTML.pfx;small" >
<!ENTITY % sub.qname     "%XHTML.pfx;sub" >
<!ENTITY % sup.qname     "%XHTML.pfx;sup" >
<!ENTITY % tt.qname      "%XHTML.pfx;tt" >

<!-- module:  xhtml-inlstruct-1.mod -->
<!ENTITY % br.qname      "%XHTML.pfx;br" >
<!ENTITY % span.qname    "%XHTML.pfx;span" >

<!-- module:  xhtml-ismap-1.mod (also csismap, ssismap) -->
<!ENTITY % map.qname     "%XHTML.pfx;map" >
<!ENTITY % area.qname    "%XHTML.pfx;area" >

<!-- module:  xhtml-link-1.mod -->
<!ENTITY % link.qname    "%XHTML.pfx;link" >

<!-- module:  xhtml-list-1.mod -->
<!ENTITY % dl.qname      "%XHTML.pfx;dl" >
<!ENTITY % dt.qname      "%XHTML.pfx;dt" >
<!ENTITY % dd.qname      "%XHTML.pfx;dd" >
<!ENTITY % ol.qname      "%XHTML.pfx;ol" >
<!ENTITY % ul.qname      "%XHTML.pfx;ul" >
<!ENTITY % li.qname      "%XHTML.pfx;li" >

<!-- module:  xhtml-meta-1.mod -->
<!ENTITY % meta.qname    "%XHTML.pfx;meta" >

<!-- module:  xhtml-param-1.mod -->
<!ENTITY % param.qname   "%XHTML.pfx;param" >

<!-- module:  xhtml-object-1.mod -->
<!ENTITY % object.qname  "%XHTML.pfx;object" >

<!-- module:  xhtml-script-1.mod -->
<!ENTITY % script.qname  "%XHTML.pfx;script" >
<!ENTITY % noscript.qname  "%XHTML.pfx;noscript" >
```

```
<!-- module:  xhtml-struct-1.mod -->
<!ENTITY % html.qname    "%XHTML.pfx;html" >
<!ENTITY % head.qname    "%XHTML.pfx;head" >
<!ENTITY % title.qname   "%XHTML.pfx;title" >
<!ENTITY % body.qname    "%XHTML.pfx;body" >

<!-- module:  xhtml-style-1.mod -->
<!ENTITY % style.qname   "%XHTML.pfx;style" >

<!-- module:  xhtml-table-1.mod -->
<!ENTITY % table.qname   "%XHTML.pfx;table" >
<!ENTITY % caption.qname "%XHTML.pfx;caption" >
<!ENTITY % thead.qname   "%XHTML.pfx;thead" >
<!ENTITY % tfoot.qname   "%XHTML.pfx;tfoot" >
<!ENTITY % tbody.qname   "%XHTML.pfx;tbody" >
<!ENTITY % colgroup.qname  "%XHTML.pfx;colgroup" >
<!ENTITY % col.qname     "%XHTML.pfx;col" >
<!ENTITY % tr.qname      "%XHTML.pfx;tr" >
<!ENTITY % th.qname      "%XHTML.pfx;th" >
<!ENTITY % td.qname      "%XHTML.pfx;td" >

<!-- module:  xhtml-ruby-1.mod -->

<!ENTITY % ruby.qname    "%XHTML.pfx;ruby" >
<!ENTITY % rbc.qname     "%XHTML.pfx;rbc" >
<!ENTITY % rtc.qname     "%XHTML.pfx;rtc" >
<!ENTITY % rb.qname      "%XHTML.pfx;rb" >
<!ENTITY % rt.qname      "%XHTML.pfx;rt" >
<!ENTITY % rp.qname      "%XHTML.pfx;rp" >

<!-- Provisional XHTML 2.0 Qualified Names  ..... -->

<!-- module:  xhtml-image-2.mod -->
<!ENTITY % alt.qname     "%XHTML.pfx;alt" >

<!-- end of xhtml-qname-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-qname-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..... -->
<!-- XHTML Embedded Object Module  ..... -->
<!-- file: xhtml-object-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
```

```
    Revision: $Id: xhtml-object-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

    This DTD module is identified by the PUBLIC and SYSTEM identifiers:

      PUBLIC "-//W3C//ELEMENTS XHTML Embedded Object 1.0//EN"
      SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-object-1.mod"

    Revisions:
    (none)
    ..... -->

<!-- Embedded Objects

       object

    This module declares the object element type and its attributes, used
    to embed external objects as part of XHTML pages. In the document,
    place param elements prior to other content within the object element.

    Note that use of this module requires instantiation of the Param
    Element Module.
-->

<!-- object: Generic Embedded Object ..... -->

<!ENTITY % object.element  "INCLUDE" >
<![%object.element;[
<!ENTITY % object.content
    "( #PCDATA | %Flow.mix; | %param.qname; )*"
>
<!ENTITY % object.qname  "object" >
<!ELEMENT %object.qname;  %object.content; >
<!-- end of object.element -->]]>

<!ENTITY % object.attlist  "INCLUDE" >
<![%object.attlist;[
<!ATTLIST %object.qname;
     %Common.attrib;
     declare      ( declare )              #IMPLIED
     classid      %URI.datatype;           #IMPLIED
     codebase     %URI.datatype;           #IMPLIED
     data         %URI.datatype;           #IMPLIED
     type         %ContentType.datatype;   #IMPLIED
     codetype     %ContentType.datatype;   #IMPLIED
     archive      %URIs.datatype;          #IMPLIED
     standby      %Text.datatype;          #IMPLIED
     height       %Length.datatype;        #IMPLIED
     width        %Length.datatype;        #IMPLIED
```

```
    name        CDATA                #IMPLIED
    tabindex    %Number.datatype;    #IMPLIED
>
<!-- end of object.attlist -->]]>

<!-- end of xhtml-object-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-object-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Server-side Image Map Module  ..................................... -->
<!-- file: xhtml-ssismap-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-ssismap-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Server-side Image Maps 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-ssismap-1.mod"

   Revisions:
#2000-10-22: added declaration for 'ismap' on <input>
   ..................................................................... -->

<!-- Server-side Image Maps

   This adds the 'ismap' attribute to the img and input elements
   to support server-side processing of a user selection.
-->

<!ATTLIST %img.qname;
    ismap       ( ismap )            #IMPLIED
>

<!ATTLIST %input.qname;
    ismap       ( ismap )            #IMPLIED
>

<!-- end of xhtml-ssismap-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-ssismap-1.mod

No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML BDO Element Module ..... -->
<!-- file: xhtml-bdo-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-bdo-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML BDO Element 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-bdo-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Bidirectional Override (bdo) Element

   This modules declares the element 'bdo', used to override the
   Unicode bidirectional algorithm for selected fragments of text.

   DEPENDENCIES:
   Relies on the conditional section keyword %XHTML.bidi; declared
   as "INCLUDE". Bidirectional text support includes both the bdo
   element and the 'dir' attribute.
-->

<!ENTITY % bdo.element  "INCLUDE" >
<![%bdo.element;[
<!ENTITY % bdo.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % bdo.qname  "bdo" >
<!ELEMENT %bdo.qname;  %bdo.content; >
<!-- end of bdo.element -->]]>

<!ENTITY % bdo.attlist  "INCLUDE" >
<![%bdo.attlist;[
<!ATTLIST %bdo.qname;
    %Core.attrib;
    xml:lang     %LanguageCode.datatype;  #IMPLIED
    dir          ( ltr | rtl )            #REQUIRED
>
]]>
```

<!-- end of xhtml-bdo-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-bdo-1.mod
No license file was found, but licenses were detected in source scan.

2002 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the MathML 2.0 DTD and
its accompanying documentation for any purpose and without fee is
hereby granted in perpetuity, provided that the above copyright notice
and this paragraph appear in all copies. The copyright holders make
no representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Revision: $Id: mathml2

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml2/mathml2-a.dtd
No license file was found, but licenses were detected in source scan.

<!--
File isoamsa.ent produced by the XSL script entities.xsl
from input data in unicode.xml.

Copyright 1998 - 2009 W3C.

Use and distribution of this code are permitted under the terms of the
W3C Software Notice and License.
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

Please report any errors to David Carlisle
via the public W3C list www-math@w3.org.

The numeric character values assigned to each entity
(should) match the Unicode assignments in Unicode 5.x.

Entity names in this file are derived from files carrying the
following notice:

(C) International Organization for Standardization 1991
Permission to copy in any form is granted for use with

conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Added Math Symbols: Arrow Relations//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isoamsa.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isoamsa PUBLIC
"-//W3C//ENTITIES Added Math Symbols: Arrow Relations//EN//XML"
"http://www.w3.org/2003/entities/2007/isoamsa.ent"
>
%isoamsa;

-->

<!ENTITY angzarr "⍼" ><!--RIGHT ANGLE WITH DOWNWARDS ZIGZAG ARROW -->
<!ENTITY cirmid "⫯" ><!--VERTICAL LINE WITH CIRCLE ABOVE -->
<!ENTITY cudarrl "⤸" ><!--RIGHT-SIDE ARC CLOCKWISE ARROW -->
<!ENTITY cudarrr "⤵" ><!--ARROW POINTING RIGHTWARDS THEN CURVING DOWNWARDS -->
<!ENTITY cularr "↶" ><!--ANTICLOCKWISE TOP SEMICIRCLE ARROW -->
<!ENTITY cularrp "⤽" ><!--TOP ARC ANTICLOCKWISE ARROW WITH PLUS -->
<!ENTITY curarr "↷" ><!--CLOCKWISE TOP SEMICIRCLE ARROW -->
<!ENTITY curarrm "⤼" ><!--TOP ARC CLOCKWISE ARROW WITH MINUS -->
<!ENTITY dArr "⇓" ><!--DOWNWARDS DOUBLE ARROW -->
<!ENTITY Darr "↡" ><!--DOWNWARDS TWO HEADED ARROW -->
<!ENTITY ddarr "⇊" ><!--DOWNWARDS PAIRED ARROWS -->
<!ENTITY DDotrahd "⤑" ><!--RIGHTWARDS ARROW WITH DOTTED STEM -->
<!ENTITY dfisht "⥿" ><!--DOWN FISH TAIL -->
<!ENTITY dHar "⥥" ><!--DOWNWARDS HARPOON WITH BARB LEFT BESIDE DOWNWARDS HARPOON WITH BARB RIGHT -->
<!ENTITY dharl "⇃" ><!--DOWNWARDS HARPOON WITH BARB LEFTWARDS -->
<!ENTITY dharr "⇂" ><!--DOWNWARDS HARPOON WITH BARB RIGHTWARDS -->
<!ENTITY duarr "⇵" ><!--DOWNWARDS ARROW LEFTWARDS OF UPWARDS ARROW -->
<!ENTITY duhar "⥯" ><!--DOWNWARDS HARPOON WITH BARB LEFT BESIDE UPWARDS HARPOON WITH BARB RIGHT -->
<!ENTITY dzigrarr "⟿" ><!--LONG RIGHTWARDS SQUIGGLE ARROW -->
<!ENTITY erarr "⥱" ><!--EQUALS SIGN ABOVE RIGHTWARDS ARROW -->
<!ENTITY harr "↔" ><!--LEFT RIGHT ARROW -->

```
<!ENTITY hArr          "&#x021D4;" ><!--LEFT RIGHT DOUBLE ARROW -->
<!ENTITY harrcir       "&#x02948;" ><!--LEFT RIGHT ARROW THROUGH SMALL CIRCLE -->
<!ENTITY harrw         "&#x021AD;" ><!--LEFT RIGHT WAVE ARROW -->
<!ENTITY hoarr         "&#x021FF;" ><!--LEFT RIGHT OPEN-HEADED ARROW -->
<!ENTITY imof          "&#x022B7;" ><!--IMAGE OF -->
<!ENTITY lAarr         "&#x021DA;" ><!--LEFTWARDS TRIPLE ARROW -->
<!ENTITY Larr          "&#x0219E;" ><!--LEFTWARDS TWO HEADED ARROW -->
<!ENTITY larrbfs       "&#x0291F;" ><!--LEFTWARDS ARROW FROM BAR TO BLACK DIAMOND -->
<!ENTITY larrfs        "&#x0291D;" ><!--LEFTWARDS ARROW TO BLACK DIAMOND -->
<!ENTITY larrhk        "&#x021A9;" ><!--LEFTWARDS ARROW WITH HOOK -->
<!ENTITY larrlp        "&#x021AB;" ><!--LEFTWARDS ARROW WITH LOOP -->
<!ENTITY larrpl        "&#x02939;" ><!--LEFT-SIDE ARC ANTICLOCKWISE ARROW -->
<!ENTITY larrsim       "&#x02973;" ><!--LEFTWARDS ARROW ABOVE TILDE OPERATOR -->
<!ENTITY larrtl        "&#x021A2;" ><!--LEFTWARDS ARROW WITH TAIL -->
<!ENTITY latail        "&#x02919;" ><!--LEFTWARDS ARROW-TAIL -->
<!ENTITY lAtail        "&#x0291B;" ><!--LEFTWARDS DOUBLE ARROW-TAIL -->
<!ENTITY lbarr         "&#x0290C;" ><!--LEFTWARDS DOUBLE DASH ARROW -->
<!ENTITY lBarr         "&#x0290E;" ><!--LEFTWARDS TRIPLE DASH ARROW -->
<!ENTITY ldca          "&#x02936;" ><!--ARROW POINTING DOWNWARDS THEN CURVING
LEFTWARDS -->
<!ENTITY ldrdhar       "&#x02967;" ><!--LEFTWARDS HARPOON WITH BARB DOWN ABOVE
RIGHTWARDS HARPOON WITH BARB DOWN -->
<!ENTITY ldrushar      "&#x0294B;" ><!--LEFT BARB DOWN RIGHT BARB UP HARPOON -->
<!ENTITY ldsh          "&#x021B2;" ><!--DOWNWARDS ARROW WITH TIP LEFTWARDS -->
<!ENTITY lfisht        "&#x0297C;" ><!--LEFT FISH TAIL -->
<!ENTITY lHar          "&#x02962;" ><!--LEFTWARDS HARPOON WITH BARB UP ABOVE LEFTWARDS
HARPOON WITH BARB DOWN -->
<!ENTITY lhard         "&#x021BD;" ><!--LEFTWARDS HARPOON WITH BARB DOWNWARDS -->
<!ENTITY lharu         "&#x021BC;" ><!--LEFTWARDS HARPOON WITH BARB UPWARDS -->
<!ENTITY lharul        "&#x0296A;" ><!--LEFTWARDS HARPOON WITH BARB UP ABOVE LONG DASH -
->
<!ENTITY llarr         "&#x021C7;" ><!--LEFTWARDS PAIRED ARROWS -->
<!ENTITY llhard        "&#x0296B;" ><!--LEFTWARDS HARPOON WITH BARB DOWN BELOW LONG
DASH -->
<!ENTITY loarr         "&#x021FD;" ><!--LEFTWARDS OPEN-HEADED ARROW -->
<!ENTITY lrarr         "&#x021C6;" ><!--LEFTWARDS ARROW OVER RIGHTWARDS ARROW -->
<!ENTITY lrhar         "&#x021CB;" ><!--LEFTWARDS HARPOON OVER RIGHTWARDS HARPOON -->
<!ENTITY lrhard        "&#x0296D;" ><!--RIGHTWARDS HARPOON WITH BARB DOWN BELOW LONG
DASH -->
<!ENTITY lsh           "&#x021B0;" ><!--UPWARDS ARROW WITH TIP LEFTWARDS -->
<!ENTITY lurdshar      "&#x0294A;" ><!--LEFT BARB UP RIGHT BARB DOWN HARPOON -->
<!ENTITY luruhar       "&#x02966;" ><!--LEFTWARDS HARPOON WITH BARB UP ABOVE
RIGHTWARDS HARPOON WITH BARB UP -->
<!ENTITY map           "&#x021A6;" ><!--RIGHTWARDS ARROW FROM BAR -->
<!ENTITY Map           "&#x02905;" ><!--RIGHTWARDS TWO-HEADED ARROW FROM BAR -->
<!ENTITY midcir        "&#x02AF0;" ><!--VERTICAL LINE WITH CIRCLE BELOW -->
<!ENTITY mumap         "&#x022B8;" ><!--MULTIMAP -->
<!ENTITY nearhk        "&#x02924;" ><!--NORTH EAST ARROW WITH HOOK -->
```

<!ENTITY nearr "↗" ><!--NORTH EAST ARROW -->
<!ENTITY neArr "⇗" ><!--NORTH EAST DOUBLE ARROW -->
<!ENTITY nesear "⤨" ><!--NORTH EAST ARROW AND SOUTH EAST ARROW -->
<!ENTITY nharr "↮" ><!--LEFT RIGHT ARROW WITH STROKE -->
<!ENTITY nhArr "⇎" ><!--LEFT RIGHT DOUBLE ARROW WITH STROKE -->
<!ENTITY nlarr "↚" ><!--LEFTWARDS ARROW WITH STROKE -->
<!ENTITY nlArr "⇍" ><!--LEFTWARDS DOUBLE ARROW WITH STROKE -->
<!ENTITY nrarr "↛" ><!--RIGHTWARDS ARROW WITH STROKE -->
<!ENTITY nrArr "⇏" ><!--RIGHTWARDS DOUBLE ARROW WITH STROKE -->
<!ENTITY nrarrc "⤳̸" ><!--WAVE ARROW POINTING DIRECTLY RIGHT with slash -->
<!ENTITY nrarrw "↝̸" ><!--RIGHTWARDS WAVE ARROW with slash -->
<!ENTITY nvHarr "⤄" ><!--LEFT RIGHT DOUBLE ARROW WITH VERTICAL STROKE -->
<!ENTITY nvlArr "⤂" ><!--LEFTWARDS DOUBLE ARROW WITH VERTICAL STROKE -->
<!ENTITY nvrArr "⤃" ><!--RIGHTWARDS DOUBLE ARROW WITH VERTICAL STROKE -->
<!ENTITY nwarhk "⤣" ><!--NORTH WEST ARROW WITH HOOK -->
<!ENTITY nwarr "↖" ><!--NORTH WEST ARROW -->
<!ENTITY nwArr "⇖" ><!--NORTH WEST DOUBLE ARROW -->
<!ENTITY nwnear "⤧" ><!--NORTH WEST ARROW AND NORTH EAST ARROW -->
<!ENTITY olarr "↺" ><!--ANTICLOCKWISE OPEN CIRCLE ARROW -->
<!ENTITY orarr "↻" ><!--CLOCKWISE OPEN CIRCLE ARROW -->
<!ENTITY origof "⊶" ><!--ORIGINAL OF -->
<!ENTITY rAarr "⇛" ><!--RIGHTWARDS TRIPLE ARROW -->
<!ENTITY Rarr "↠" ><!--RIGHTWARDS TWO HEADED ARROW -->
<!ENTITY rarrap "⥵" ><!--RIGHTWARDS ARROW ABOVE ALMOST EQUAL TO -->
<!ENTITY rarrbfs "⤠" ><!--RIGHTWARDS ARROW FROM BAR TO BLACK DIAMOND -->
<!ENTITY rarrc "⤳" ><!--WAVE ARROW POINTING DIRECTLY RIGHT -->
<!ENTITY rarrfs "⤞" ><!--RIGHTWARDS ARROW TO BLACK DIAMOND -->
<!ENTITY rarrhk "↪" ><!--RIGHTWARDS ARROW WITH HOOK -->
<!ENTITY rarrlp "↬" ><!--RIGHTWARDS ARROW WITH LOOP -->
<!ENTITY rarrpl "⥅" ><!--RIGHTWARDS ARROW WITH PLUS BELOW -->
<!ENTITY rarrsim "⥴" ><!--RIGHTWARDS ARROW ABOVE TILDE OPERATOR -->
<!ENTITY rarrtl "↣" ><!--RIGHTWARDS ARROW WITH TAIL -->
<!ENTITY Rarrtl "⤖" ><!--RIGHTWARDS TWO-HEADED ARROW WITH TAIL -->
<!ENTITY rarrw "↝" ><!--RIGHTWARDS WAVE ARROW -->
<!ENTITY ratail "⤚" ><!--RIGHTWARDS ARROW-TAIL -->
<!ENTITY rAtail "⤜" ><!--RIGHTWARDS DOUBLE ARROW-TAIL -->
<!ENTITY rbarr "⤍" ><!--RIGHTWARDS DOUBLE DASH ARROW -->
<!ENTITY rBarr "⤏" ><!--RIGHTWARDS TRIPLE DASH ARROW -->
<!ENTITY RBarr "⤐" ><!--RIGHTWARDS TWO-HEADED TRIPLE DASH ARROW -->
<!ENTITY rdca "⤷" ><!--ARROW POINTING DOWNWARDS THEN CURVING RIGHTWARDS -->
<!ENTITY rdldhar "⥩" ><!--RIGHTWARDS HARPOON WITH BARB DOWN ABOVE LEFTWARDS HARPOON WITH BARB DOWN -->
<!ENTITY rdsh "↳" ><!--DOWNWARDS ARROW WITH TIP RIGHTWARDS -->
<!ENTITY rfisht "⥽" ><!--RIGHT FISH TAIL -->
<!ENTITY rHar "⥤" ><!--RIGHTWARDS HARPOON WITH BARB UP ABOVE RIGHTWARDS HARPOON WITH BARB DOWN -->

<!ENTITY rhard "⇁" ><!--RIGHTWARDS HARPOON WITH BARB DOWNWARDS -->
<!ENTITY rharu "⇀" ><!--RIGHTWARDS HARPOON WITH BARB UPWARDS -->
<!ENTITY rharul "⥬" ><!--RIGHTWARDS HARPOON WITH BARB UP ABOVE LONG DASH -->
<!ENTITY rlarr "⇄" ><!--RIGHTWARDS ARROW OVER LEFTWARDS ARROW -->
<!ENTITY rlhar "⇌" ><!--RIGHTWARDS HARPOON OVER LEFTWARDS HARPOON -->
<!ENTITY roarr "⇾" ><!--RIGHTWARDS OPEN-HEADED ARROW -->
<!ENTITY rrarr "⇉" ><!--RIGHTWARDS PAIRED ARROWS -->
<!ENTITY rsh "↱" ><!--UPWARDS ARROW WITH TIP RIGHTWARDS -->
<!ENTITY ruluhar "⥨" ><!--RIGHTWARDS HARPOON WITH BARB UP ABOVE LEFTWARDS HARPOON WITH BARB UP -->
<!ENTITY searhk "⤥" ><!--SOUTH EAST ARROW WITH HOOK -->
<!ENTITY searr "↘" ><!--SOUTH EAST ARROW -->
<!ENTITY seArr "⇘" ><!--SOUTH EAST DOUBLE ARROW -->
<!ENTITY seswar "⤩" ><!--SOUTH EAST ARROW AND SOUTH WEST ARROW -->
<!ENTITY simrarr "⥲" ><!--TILDE OPERATOR ABOVE RIGHTWARDS ARROW -->
<!ENTITY slarr "←" ><!--LEFTWARDS ARROW -->
<!ENTITY srarr "→" ><!--RIGHTWARDS ARROW -->
<!ENTITY swarhk "⤦" ><!--SOUTH WEST ARROW WITH HOOK -->
<!ENTITY swarr "↙" ><!--SOUTH WEST ARROW -->
<!ENTITY swArr "⇙" ><!--SOUTH WEST DOUBLE ARROW -->
<!ENTITY swnwar "⤪" ><!--SOUTH WEST ARROW AND NORTH WEST ARROW -->
<!ENTITY uArr "⇑" ><!--UPWARDS DOUBLE ARROW -->
<!ENTITY Uarr "↟" ><!--UPWARDS TWO HEADED ARROW -->
<!ENTITY Uarrocir "⥉" ><!--UPWARDS TWO-HEADED ARROW FROM SMALL CIRCLE -->
<!ENTITY udarr "⇅" ><!--UPWARDS ARROW LEFTWARDS OF DOWNWARDS ARROW -->
<!ENTITY udhar "⥮" ><!--UPWARDS HARPOON WITH BARB LEFT BESIDE DOWNWARDS HARPOON WITH BARB RIGHT -->
<!ENTITY ufisht "⥾" ><!--UP FISH TAIL -->
<!ENTITY uHar "⥣" ><!--UPWARDS HARPOON WITH BARB LEFT BESIDE UPWARDS HARPOON WITH BARB RIGHT -->
<!ENTITY uharl "↿" ><!--UPWARDS HARPOON WITH BARB LEFTWARDS -->
<!ENTITY uharr "↾" ><!--UPWARDS HARPOON WITH BARB RIGHTWARDS -->
<!ENTITY uuarr "⇈" ><!--UPWARDS PAIRED ARROWS -->
<!ENTITY varr "↕" ><!--UP DOWN ARROW -->
<!ENTITY vArr "⇕" ><!--UP DOWN DOUBLE ARROW -->
<!ENTITY xharr "⟷" ><!--LONG LEFT RIGHT ARROW -->
<!ENTITY xhArr "⟺" ><!--LONG LEFT RIGHT DOUBLE ARROW -->
<!ENTITY xlarr "⟵" ><!--LONG LEFTWARDS ARROW -->
<!ENTITY xlArr "⟸" ><!--LONG LEFTWARDS DOUBLE ARROW -->
<!ENTITY xmap "⟼" ><!--LONG RIGHTWARDS ARROW FROM BAR -->
<!ENTITY xrarr "⟶" ><!--LONG RIGHTWARDS ARROW -->
<!ENTITY xrArr "⟹" ><!--LONG RIGHTWARDS DOUBLE ARROW -->
<!ENTITY zigrarr "⇝" ><!--RIGHTWARDS SQUIGGLE ARROW -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-

jar/net/sf/saxon/data/w3c/mathml/mathml3/isoamsa.ent
No license file was found, but licenses were detected in source scan.

```
<!--
   File mmlextra.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

-->


<!--

     Public identifier: -//W3C//ENTITIES Additional MathML Symbols//EN//XML
     System identifier: http://www.w3.org/2003/entities/2007/mmlextra.ent

   The public identifier should always be used verbatim.
   The system identifier may be changed to suit local requirements.

   Typical invocation:

     <!ENTITY % mmlextra PUBLIC
       "-//W3C//ENTITIES Additional MathML Symbols//EN//XML"
       "http://www.w3.org/2003/entities/2007/mmlextra.ent"
     >
     %mmlextra;

-->

<!ENTITY af             "&#x02061;" ><!--FUNCTION APPLICATION -->
<!ENTITY aopf           "&#x1D552;" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL A -->
<!ENTITY asympeq        "&#x0224D;" ><!--EQUIVALENT TO -->
<!ENTITY bopf           "&#x1D553;" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL B -->
<!ENTITY copf           "&#x1D554;" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL C -->
<!ENTITY Cross          "&#x02A2F;" ><!--VECTOR OR CROSS PRODUCT -->
<!ENTITY dd             "&#x02146;" ><!--DOUBLE-STRUCK ITALIC SMALL D -->
<!ENTITY DD             "&#x02145;" ><!--DOUBLE-STRUCK ITALIC CAPITAL D -->
```

<!ENTITY dopf "𝕕" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL D -->
<!ENTITY DownArrowBar "⤓" ><!--DOWNWARDS ARROW TO BAR -->
<!ENTITY DownBreve " ̑" ><!--COMBINING INVERTED BREVE -->
<!ENTITY DownLeftRightVector "⥐" ><!--LEFT BARB DOWN RIGHT BARB DOWN HARPOON -->
<!ENTITY DownLeftTeeVector "⥞" ><!--LEFTWARDS HARPOON WITH BARB DOWN FROM BAR -->
<!ENTITY DownLeftVectorBar "⥖" ><!--LEFTWARDS HARPOON WITH BARB DOWN TO BAR -->
<!ENTITY DownRightTeeVector "⥟" ><!--RIGHTWARDS HARPOON WITH BARB DOWN FROM BAR -->
<!ENTITY DownRightVectorBar "⥗" ><!--RIGHTWARDS HARPOON WITH BARB DOWN TO BAR -->
<!ENTITY ee "ⅇ" ><!--DOUBLE-STRUCK ITALIC SMALL E -->
<!ENTITY EmptySmallSquare "◻" ><!--WHITE MEDIUM SQUARE -->
<!ENTITY EmptyVerySmallSquare "▫" ><!--WHITE SMALL SQUARE -->
<!ENTITY eopf "𝕖" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL E -->
<!ENTITY Equal "⩵" ><!--TWO CONSECUTIVE EQUALS SIGNS -->
<!ENTITY FilledSmallSquare "◼" ><!--BLACK MEDIUM SQUARE -->
<!ENTITY FilledVerySmallSquare "▪" ><!--BLACK SMALL SQUARE -->
<!ENTITY fopf "𝕗" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL F -->
<!ENTITY gopf "𝕘" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL G -->
<!ENTITY GreaterGreater "⪢" ><!--DOUBLE NESTED GREATER-THAN -->
<!ENTITY Hat "^" ><!--CIRCUMFLEX ACCENT -->
<!ENTITY hopf "𝕙" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL H -->
<!ENTITY HorizontalLine "─" ><!--BOX DRAWINGS LIGHT HORIZONTAL -->
<!ENTITY ic "⁣" ><!--INVISIBLE SEPARATOR -->
<!ENTITY ii "ⅈ" ><!--DOUBLE-STRUCK ITALIC SMALL I -->
<!ENTITY iopf "𝕚" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL I -->
<!ENTITY it "⁢" ><!--INVISIBLE TIMES -->
<!ENTITY jopf "𝕛" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL J -->
<!ENTITY kopf "𝕜" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL K -->
<!ENTITY larrb "⇤" ><!--LEFTWARDS ARROW TO BAR -->
<!ENTITY LeftDownTeeVector "⥡" ><!--DOWNWARDS HARPOON WITH BARB LEFT FROM BAR -->
<!ENTITY LeftDownVectorBar "⥙" ><!--DOWNWARDS HARPOON WITH BARB LEFT TO BAR -->
<!ENTITY LeftRightVector "⥎" ><!--LEFT BARB UP RIGHT BARB UP HARPOON -->
<!ENTITY LeftTeeVector "⥚" ><!--LEFTWARDS HARPOON WITH BARB UP FROM BAR -->
<!ENTITY LeftTriangleBar "⧏" ><!--LEFT TRIANGLE BESIDE VERTICAL BAR -->
<!ENTITY LeftUpDownVector "⥑" ><!--UP BARB LEFT DOWN BARB LEFT HARPOON -->
<!ENTITY LeftUpTeeVector "⥠" ><!--UPWARDS HARPOON WITH BARB LEFT FROM BAR -->
<!ENTITY LeftUpVectorBar "⥘" ><!--UPWARDS HARPOON WITH BARB LEFT TO BAR -->
<!ENTITY LeftVectorBar "⥒" ><!--LEFTWARDS HARPOON WITH BARB UP TO BAR -->
<!ENTITY LessLess "⪡" ><!--DOUBLE NESTED LESS-THAN -->
<!ENTITY lopf "𝕝" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL L -->
<!ENTITY mapstodown "↧" ><!--DOWNWARDS ARROW FROM BAR -->
<!ENTITY mapstoleft "↤" ><!--LEFTWARDS ARROW FROM BAR -->
<!ENTITY mapstoup "↥" ><!--UPWARDS ARROW FROM BAR -->
<!ENTITY MediumSpace " " ><!--MEDIUM MATHEMATICAL SPACE -->
<!ENTITY mopf "𝕞" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL M -->

<!ENTITY nbump "≎̸" ><!--GEOMETRICALLY EQUIVALENT TO with slash -->
<!ENTITY nbumpe "≏̸" ><!--DIFFERENCE BETWEEN with slash -->
<!ENTITY nesim "≂̸" ><!--MINUS TILDE with slash -->
<!ENTITY NewLine "
" ><!--LINE FEED (LF) -->
<!ENTITY NoBreak "" ><!--WORD JOINER -->
<!ENTITY nopf "𝕟" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL N -->
<!ENTITY NotCupCap "≭" ><!--NOT EQUIVALENT TO -->
<!ENTITY NotHumpEqual "≏̸" ><!--DIFFERENCE BETWEEN with slash -->
<!ENTITY NotLeftTriangleBar "⧏̸" ><!--LEFT TRIANGLE BESIDE VERTICAL BAR with slash -->
<!ENTITY NotNestedGreaterGreater "⪢̸" ><!--DOUBLE NESTED GREATER-THAN with slash -->
<!ENTITY NotNestedLessLess "⪡̸" ><!--DOUBLE NESTED LESS-THAN with slash -->
<!ENTITY NotRightTriangleBar "⧐̸" ><!--VERTICAL BAR BESIDE RIGHT TRIANGLE with slash -->
<!ENTITY NotSquareSubset "⊏̸" ><!--SQUARE IMAGE OF with slash -->
<!ENTITY NotSquareSuperset "⊐̸" ><!--SQUARE ORIGINAL OF with slash -->
<!ENTITY NotSucceedsTilde "≿̸" ><!--SUCCEEDS OR EQUIVALENT TO with slash -->
<!ENTITY oopf "𝕠" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL O -->
<!ENTITY OverBar "‾" ><!--OVERLINE -->
<!ENTITY OverBrace "⏞" ><!--TOP CURLY BRACKET -->
<!ENTITY OverBracket "⎴" ><!--TOP SQUARE BRACKET -->
<!ENTITY OverParenthesis "⏜" ><!--TOP PARENTHESIS -->
<!ENTITY planckh "ℎ" ><!--PLANCK CONSTANT -->
<!ENTITY popf "𝕡" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL P -->
<!ENTITY Product "∏" ><!--N-ARY PRODUCT -->
<!ENTITY qopf "𝕢" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL Q -->
<!ENTITY rarrb "⇥" ><!--RIGHTWARDS ARROW TO BAR -->
<!ENTITY RightDownTeeVector "⥝" ><!--DOWNWARDS HARPOON WITH BARB RIGHT FROM BAR -->
<!ENTITY RightDownVectorBar "⥕" ><!--DOWNWARDS HARPOON WITH BARB RIGHT TO BAR -->
<!ENTITY RightTeeVector "⥛" ><!--RIGHTWARDS HARPOON WITH BARB UP FROM BAR -->
<!ENTITY RightTriangleBar "⧐" ><!--VERTICAL BAR BESIDE RIGHT TRIANGLE -->
<!ENTITY RightUpDownVector "⥏" ><!--UP BARB RIGHT DOWN BARB RIGHT HARPOON -->
<!ENTITY RightUpTeeVector "⥜" ><!--UPWARDS HARPOON WITH BARB RIGHT FROM BAR -->
<!ENTITY RightUpVectorBar "⥔" ><!--UPWARDS HARPOON WITH BARB RIGHT TO BAR -->
<!ENTITY RightVectorBar "⥓" ><!--RIGHTWARDS HARPOON WITH BARB UP TO BAR -->
<!ENTITY ropf "𝕣" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL R -->
<!ENTITY RoundImplies "⥰" ><!--RIGHT DOUBLE ARROW WITH ROUNDED HEAD -->
<!ENTITY RuleDelayed "⧴" ><!--RULE-DELAYED -->
<!ENTITY sopf "𝕤" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL S -->
<!ENTITY Tab "	" ><!--CHARACTER TABULATION -->
<!ENTITY ThickSpace " " ><!--space of width 5/18 em -->
<!ENTITY topf "𝕥" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL T -->
<!ENTITY UnderBar "_" ><!--LOW LINE -->
<!ENTITY UnderBrace "⏟" ><!--BOTTOM CURLY BRACKET -->
<!ENTITY UnderBracket "⎵" ><!--BOTTOM SQUARE BRACKET -->

<!ENTITY UnderParenthesis "⏝" ><!--BOTTOM PARENTHESIS -->
<!ENTITY uopf "𝕦" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL U -->
<!ENTITY UpArrowBar "⤒" ><!--UPWARDS ARROW TO BAR -->
<!ENTITY Upsilon "Υ" ><!--GREEK CAPITAL LETTER UPSILON -->
<!ENTITY VerticalLine "|" ><!--VERTICAL LINE -->
<!ENTITY VerticalSeparator "❘" ><!--LIGHT VERTICAL BAR -->
<!ENTITY vopf "𝕧" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL V -->
<!ENTITY wopf "𝕨" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL W -->
<!ENTITY xopf "𝕩" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL X -->
<!ENTITY yopf "𝕪" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL Y -->
<!ENTITY ZeroWidthSpace "" ><!--ZERO WIDTH SPACE -->
<!ENTITY zopf "𝕫" ><!--MATHEMATICAL DOUBLE-STRUCK SMALL Z -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/mmlextra.ent
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Basic Document Model Module ..... -->
<!-- file: svg11-basic-model.mod

This is SVG Basic, a proper subset of SVG.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg11-basic-model.mod,v 1.5 2002/11/14 15:11:03 fujisawa Exp $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Basic Document Model//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11-basic-model.mod"

..... -->

<!-- SVG 1.1 Basic Document Model

This module describes the groupings of elements that make up
common content models for SVG elements.
-->

<!-- Content Models (Default) ..... -->

<!ENTITY % SVG.Marker.class "" >
<!ENTITY % SVG.Cursor.class "" >

<!-- module: svg-structure.mod ..... -->

<!ENTITY % SVG.Description.extra.class "" >
<!ENTITY % SVG.Description.class

```
  "%SVG.desc.qname; | %SVG.title.qname; | %SVG.metadata.qname;
   %SVG.Description.extra.class;"
>

<!ENTITY % SVG.Use.extra.class "" >
<!ENTITY % SVG.Use.class
  "| %SVG.use.qname; %SVG.Use.extra.class;"
>

<!ENTITY % SVG.Structure.extra.class "" >
<!ENTITY % SVG.Structure.class
  "| %SVG.svg.qname; | %SVG.g.qname; | %SVG.defs.qname; | %SVG.symbol.qname;
    %SVG.Use.class; %SVG.Structure.extra.class;"
>

<!-- module: svg-conditional.mod ..... -->

<!ENTITY % SVG.Conditional.extra.class "" >
<!ENTITY % SVG.Conditional.class
  "| %SVG.switch.qname; %SVG.Conditional.extra.class;"
>

<!-- module: svg-image.mod ..... -->

<!ENTITY % SVG.Image.extra.class "" >
<!ENTITY % SVG.Image.class
  "| %SVG.image.qname; %SVG.Image.extra.class;"
>

<!-- module: svg-style.mod ..... -->

<!ENTITY % SVG.Style.extra.class "" >
<!ENTITY % SVG.Style.class
  "| %SVG.style.qname; %SVG.Style.extra.class;"
>

<!-- module: svg-shape.mod ..... -->

<!ENTITY % SVG.Shape.extra.class "" >
<!ENTITY % SVG.Shape.class
  "| %SVG.path.qname; | %SVG.rect.qname; | %SVG.circle.qname;
   | %SVG.line.qname; | %SVG.ellipse.qname; | %SVG.polyline.qname;
   | %SVG.polygon.qname; %SVG.Shape.extra.class;"
>

<!-- module: svg-text.mod ..... -->

<!ENTITY % SVG.Text.extra.class "" >
```

```
<!ENTITY % SVG.Text.class
  "| %SVG.text.qname; | %SVG.altGlyphDef.qname; %SVG.Text.extra.class;"
>

<!ENTITY % SVG.TextContent.extra.class "" >
<!ENTITY % SVG.TextContent.class
  "| %SVG.tspan.qname; | %SVG.tref.qname; | %SVG.textPath.qname;
   | %SVG.altGlyph.qname; %SVG.TextContent.extra.class;"
>

<!-- module: svg-profile.mod ..... -->

<!ENTITY % SVG.ColorProfile.extra.class "" >
<!ENTITY % SVG.ColorProfile.class
  "| %SVG.color-profile.qname; %SVG.ColorProfile.extra.class;"
>

<!-- module: svg-gradient.mod ..... -->

<!ENTITY % SVG.Gradient.extra.class "" >
<!ENTITY % SVG.Gradient.class
  "| %SVG.linearGradient.qname; | %SVG.radialGradient.qname;
    %SVG.Gradient.extra.class;"
>

<!-- module: svg-pattern.mod ..... -->

<!ENTITY % SVG.Pattern.extra.class "" >
<!ENTITY % SVG.Pattern.class
  "| %SVG.pattern.qname; %SVG.Pattern.extra.class;"
>

<!-- module: svg-basic-clip.mod ..... -->

<!ENTITY % SVG.Clip.extra.class "" >
<!ENTITY % SVG.Clip.class
  "| %SVG.clipPath.qname; %SVG.Clip.extra.class;"
>

<!-- module: svg-mask.mod ..... -->

<!ENTITY % SVG.Mask.extra.class "" >
<!ENTITY % SVG.Mask.class
  "| %SVG.mask.qname; %SVG.Mask.extra.class;"
>

<!-- module: svg-basic-filter.mod ..... -->
```

```
<!ENTITY % SVG.Filter.extra.class "" >
<!ENTITY % SVG.Filter.class
  "| %SVG.filter.qname; %SVG.Filter.extra.class;"
>

<!ENTITY % SVG.FilterPrimitive.extra.class "" >
<!ENTITY % SVG.FilterPrimitive.class
  "| %SVG.feBlend.qname; | %SVG.feColorMatrix.qname;
   | %SVG.feComponentTransfer.qname; | %SVG.feComposite.qname;
   | %SVG.feFlood.qname; | %SVG.feGaussianBlur.qname; | %SVG.feImage.qname;
   | %SVG.feMerge.qname; | %SVG.feOffset.qname; | %SVG.feTile.qname;
     %SVG.FilterPrimitive.extra.class;"
>

<!-- module: svg-hyperlink.mod ..... -->

<!ENTITY % SVG.Hyperlink.extra.class "" >
<!ENTITY % SVG.Hyperlink.class
  "| %SVG.a.qname; %SVG.Hyperlink.extra.class;"
>

<!-- module: svg-view.mod ..... -->

<!ENTITY % SVG.View.extra.class "" >
<!ENTITY % SVG.View.class
  "| %SVG.view.qname; %SVG.View.extra.class;"
>

<!-- module: svg-script.mod ..... -->

<!ENTITY % SVG.Script.extra.class "" >
<!ENTITY % SVG.Script.class
  "| %SVG.script.qname; %SVG.Script.extra.class;"
>

<!-- module: svg-animation.mod ..... -->

<!ENTITY % SVG.Animation.extra.class "" >
<!ENTITY % SVG.Animation.class
  "%SVG.animate.qname; | %SVG.set.qname; | %SVG.animateMotion.qname; |
   %SVG.animateColor.qname; | %SVG.animateTransform.qname;
   %SVG.Animation.extra.class;"
>

<!-- module: svg-font.mod ..... -->

<!ENTITY % SVG.Font.extra.class "" >
<!ENTITY % SVG.Font.class
```

```
  "| %SVG.font.qname; | %SVG.font-face.qname; %SVG.Font.extra.class;"
>

<!-- module: svg-extensibility.mod ..... -->

<!ENTITY % SVG.Extensibility.extra.class "" >
<!ENTITY % SVG.Extensibility.class
  "| %SVG.foreignObject.qname; %SVG.Extensibility.extra.class;"
>

<!-- end of svg11-basic-model.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg11-basic-model.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..... -->
<!-- XHTML Simplified Forms Module  ..... -->
<!-- file: xhtml-basic-form-1.mod

   This is XHTML Basic, a proper subset of XHTML.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-basic-form-1.mod,v 4.1 2001/04/05 06:57:40 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Basic Forms 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-basic-form-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Basic Forms

   This forms module is based on the HTML 3.2 forms model, with
   the WAI-requested addition of the label element. While this
   module essentially mimics the content model and attributes of
   HTML 3.2 forms, the element types declared herein also include
   all HTML 4 common attributes.

      form, label, input, select, option, textarea
-->

<!-- declare qualified element type names:
-->
<!ENTITY % form.qname  "form" >
```

```
<!ENTITY % label.qname  "label" >
<!ENTITY % input.qname  "input" >
<!ENTITY % select.qname  "select" >
<!ENTITY % option.qname  "option" >
<!ENTITY % textarea.qname  "textarea" >

<!-- %BlkNoForm.mix; includes all non-form block elements,
   plus %Misc.class;
-->
<!ENTITY % BlkNoForm.mix
   "%Heading.class;
    | %List.class;
    | %BlkStruct.class;
    %BlkPhras.class;
    %BlkPres.class;
    | %table.qname;
    %Block.extra;
    %Misc.class;"
>

<!-- form: Form Element ..... -->

<!ENTITY % form.element  "INCLUDE" >
<![%form.element;[
<!ENTITY % form.content
   "( %BlkNoForm.mix; )+"
>
<!ELEMENT %form.qname;  %form.content; >
<!-- end of form.element -->]]>

<!ENTITY % form.attlist  "INCLUDE" >
<![%form.attlist;[
<!ATTLIST %form.qname;
    %Common.attrib;
    action      %URI.datatype;          #REQUIRED
    method      ( get | post )          'get'
    enctype     %ContentType.datatype;   'application/x-www-form-urlencoded'
>
<!-- end of form.attlist -->]]>

<!-- label: Form Field Label Text ..... -->

<!ENTITY % label.element  "INCLUDE" >
<![%label.element;[
<!-- Each label must not contain more than ONE field
-->
<!ENTITY % label.content
   "( #PCDATA
```

```
    | %input.qname; | %select.qname; | %textarea.qname;
    | %InlStruct.class;
    %InlPhras.class;
    %I18n.class;
    %InlPres.class;
    %InlSpecial.class;
    %Misc.class; )*"
>
<!ELEMENT %label.qname;  %label.content; >
<!-- end of label.element -->]]>

<!ENTITY % label.attlist  "INCLUDE" >
<![%label.attlist;[
<!ATTLIST %label.qname;
    %Common.attrib;
    for         IDREF                #IMPLIED
    accesskey   %Character.datatype;    #IMPLIED
>
<!-- end of label.attlist -->]]>

<!-- input: Form Control ..... -->

<!ENTITY % input.element  "INCLUDE" >
<![%input.element;[
<!ENTITY % input.content  "EMPTY" >
<!ELEMENT %input.qname;  %input.content; >
<!-- end of input.element -->]]>

<!-- Basic Forms removes 'button', 'image' and 'file' input types.
-->
<!ENTITY % input.attlist  "INCLUDE" >
<![%input.attlist;[
<!ENTITY % InputType.class
   "( text | password | checkbox | radio
    | submit | reset | hidden )"
>
<!-- attribute name required for all but submit & reset
-->
<!ATTLIST %input.qname;
    %Common.attrib;
    type        %InputType.class;       'text'
    name        CDATA                   #IMPLIED
    value       CDATA                   #IMPLIED
    checked     ( checked )             #IMPLIED
    size        %Number.datatype;       #IMPLIED
    maxlength   %Number.datatype;       #IMPLIED
    src         %URI.datatype;          #IMPLIED
    tabindex    %Number.datatype;       #IMPLIED
```

```
    accesskey    %Character.datatype;     #IMPLIED
>
<!-- end of input.attlist -->]]>

<!-- select: Option Selector ..... -->

<!ENTITY % select.element  "INCLUDE" >
<![%select.element;[
<!ENTITY % select.content  "( %option.qname; )+" >
<!ELEMENT %select.qname;  %select.content; >
<!-- end of select.element -->]]>

<!ENTITY % select.attlist  "INCLUDE" >
<![%select.attlist;[
<!ATTLIST %select.qname;
    %Common.attrib;
    name         CDATA                  #IMPLIED
    size         %Number.datatype;      #IMPLIED
    multiple     ( multiple )           #IMPLIED
    tabindex     %Number.datatype;      #IMPLIED
>
<!-- end of select.attlist -->]]>

<!-- option: Selectable Choice ..... -->

<!ENTITY % option.element  "INCLUDE" >
<![%option.element;[
<!ENTITY % option.content  "( #PCDATA )" >
<!ELEMENT %option.qname;  %option.content; >
<!-- end of option.element -->]]>

<!ENTITY % option.attlist  "INCLUDE" >
<![%option.attlist;[
<!ATTLIST %option.qname;
    %Common.attrib;
    selected     ( selected )           #IMPLIED
    value        CDATA                  #IMPLIED
>
<!-- end of option.attlist -->]]>

<!-- textarea: Multi-Line Text Field ..... -->

<!ENTITY % textarea.element  "INCLUDE" >
<![%textarea.element;[
<!ENTITY % textarea.content  "( #PCDATA )" >
<!ELEMENT %textarea.qname;  %textarea.content; >
<!-- end of textarea.element -->]]>
```

```
<!ENTITY % textarea.attlist  "INCLUDE" >
<![%textarea.attlist;[
<!ATTLIST %textarea.qname;
    %Common.attrib;
    name        CDATA                  #IMPLIED
    rows        %Number.datatype;      #REQUIRED
    cols        %Number.datatype;      #REQUIRED
    tabindex    %Number.datatype;      #IMPLIED
    accesskey   %Character.datatype;   #IMPLIED
>
<!-- end of textarea.attlist -->]]>

<!-- end of xhtml-basic-form-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-basic-form-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Hypertext Module  ................................................ -->
<!-- file: xhtml-hypertext-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-hypertext-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Hypertext 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-hypertext-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Hypertext

      a

   This module declares the anchor ('a') element type, which
   defines the source of a hypertext link. The destination
   (or link 'target') is identified via its 'id' attribute
   rather than the 'name' attribute as was used in HTML.
-->

<!-- .............  Anchor Element  ............. -->
```

```
<!ENTITY % a.element  "INCLUDE" >
<![%a.element;[
<!ENTITY % a.content
   "( #PCDATA | %InlNoAnchor.mix; )*"
>
<!ENTITY % a.qname  "a" >
<!ELEMENT %a.qname;  %a.content; >
<!-- end of a.element -->]]>

<!ENTITY % a.attlist  "INCLUDE" >
<![%a.attlist;[
<!ATTLIST %a.qname;
    %Common.attrib;
    href         %URI.datatype;           #IMPLIED
    charset      %Charset.datatype;       #IMPLIED
    type         %ContentType.datatype;   #IMPLIED
    hreflang     %LanguageCode.datatype;  #IMPLIED
    rel          %LinkTypes.datatype;     #IMPLIED
    rev          %LinkTypes.datatype;     #IMPLIED
    accesskey    %Character.datatype;     #IMPLIED
    tabindex     %Number.datatype;        #IMPLIED
>
<!-- end of a.attlist -->]]>

<!-- end of xhtml-hypertext-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-hypertext-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Block Presentation Module  ....................................... -->
<!-- file: xhtml-blkpres-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-blkpres-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Block Presentation 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-blkpres-1.mod"

   Revisions:
   (none)
   ....................................................................... -->
```

```
<!-- Block Presentational Elements

     hr

   This module declares the elements and their attributes used to
   support block-level presentational markup.
-->

<!ENTITY % hr.element  "INCLUDE" >
<![%hr.element;[
<!ENTITY % hr.content  "EMPTY" >
<!ENTITY % hr.qname  "hr" >
<!ELEMENT %hr.qname;  %hr.content; >
<!-- end of hr.element -->]]>

<!ENTITY % hr.attlist  "INCLUDE" >
<![%hr.attlist;[
<!ATTLIST %hr.qname;
    %Common.attrib;
>
<!-- end of hr.attlist -->]]>

<!-- end of xhtml-blkpres-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-blkpres-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- XHTML Modular Framework Module  ..................................... -->
<!-- file: xhtml-framework-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-framework-1.mod,v 1.3 2004/09/07 12:12:18 dom Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Modular Framework 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-framework-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Modular Framework
```

```
   This required module instantiates the modules needed
   to support the XHTML modularization model, including:

     + notations
     + datatypes
     + namespace-qualified names
     + common attributes
     + document model
     + character entities

   The Intrinsic Events module is ignored by default but
   occurs in this module because it must be instantiated
   prior to Attributes but after Datatypes.
-->

<!ENTITY % xhtml-arch.module "INCLUDE" >
<![%xhtml-arch.module;[
<!ENTITY % xhtml-arch.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Base Architecture 1.0//EN"
         "xhtml-arch-1.mod" >
%xhtml-arch.mod;]]>

<!ENTITY % xhtml-notations.module "INCLUDE" >
<![%xhtml-notations.module;[
<!ENTITY % xhtml-notations.mod
   PUBLIC "-//W3C//NOTATIONS XHTML Notations 1.0//EN"
         "xhtml-notations-1.mod" >
%xhtml-notations.mod;]]>

<!ENTITY % xhtml-datatypes.module "INCLUDE" >
<![%xhtml-datatypes.module;[
<!ENTITY % xhtml-datatypes.mod
   PUBLIC "-//W3C//ENTITIES XHTML Datatypes 1.0//EN"
         "xhtml-datatypes-1.mod" >
%xhtml-datatypes.mod;]]>

<!-- placeholder for XLink support module -->
<!ENTITY % xhtml-xlink.mod "" >
%xhtml-xlink.mod;

<!ENTITY % xhtml-qname.module "INCLUDE" >
<![%xhtml-qname.module;[
<!ENTITY % xhtml-qname.mod
   PUBLIC "-//W3C//ENTITIES XHTML Qualified Names 1.0//EN"
         "xhtml-qname-1.mod" >
%xhtml-qname.mod;]]>

<!ENTITY % xhtml-events.module "IGNORE" >
```

```
<![%xhtml-events.module;[
<!ENTITY % xhtml-events.mod
   PUBLIC "-//W3C//ENTITIES XHTML Intrinsic Events 1.0//EN"
         "xhtml-events-1.mod" >
%xhtml-events.mod;]]>

<!ENTITY % xhtml-attribs.module "INCLUDE" >
<![%xhtml-attribs.module;[
<!ENTITY % xhtml-attribs.mod
   PUBLIC "-//W3C//ENTITIES XHTML Common Attributes 1.0//EN"
         "xhtml-attribs-1.mod" >
%xhtml-attribs.mod;]]>

<!-- placeholder for content model redeclarations -->
<!ENTITY % xhtml-model.redecl "" >
%xhtml-model.redecl;

<!ENTITY % xhtml-model.module "INCLUDE" >
<![%xhtml-model.module;[
<!-- instantiate the Document Model module declared in the DTD driver
-->
%xhtml-model.mod;]]>

<!ENTITY % xhtml-charent.module "INCLUDE" >
<![%xhtml-charent.module;[
<!ENTITY % xhtml-charent.mod
   PUBLIC "-//W3C//ENTITIES XHTML Character Entities 1.0//EN"
         "xhtml-charent-1.mod" >
%xhtml-charent.mod;]]>

<!-- end of xhtml-framework-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-framework-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Basic Clip Module ..... -->
<!-- file: svg-basic-clip.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-basic-clip.mod,v 1.4 2002/11/03 15:54:14 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Basic Clip//EN"
```

```
      SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-clip.mod"

    ..... -->

<!-- Basic Clip

      clipPath

    This module declares markup to provide support for clipping.
-->

<!-- 'clip-path' property/attribute value (e.g., 'none', <uri>) -->
<!ENTITY % ClipPathValue.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.clipPath.qname "clipPath" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Clip.class ..... -->

<!ENTITY % SVG.Clip.extra.class "" >

<!ENTITY % SVG.Clip.class
   "| %SVG.clipPath.qname; %SVG.Clip.extra.class;"
>

<!-- SVG.Clip.attrib ..... -->

<!ENTITY % SVG.Clip.extra.attrib "" >

<!ENTITY % SVG.Clip.attrib
```

```
  "clip-path %ClipPathValue.datatype; #IMPLIED
   clip-rule %ClipFillRule.datatype; #IMPLIED
   %SVG.Clip.extra.attrib;"
>

<!-- clipPath: Clip Path Element ..... -->

<!ENTITY % SVG.clipPath.extra.content "" >

<!ENTITY % SVG.clipPath.element "INCLUDE" >
<![%SVG.clipPath.element;[
<!ENTITY % SVG.clipPath.content
  "(( %SVG.Description.class; )*, (( %SVG.Animation.class; %SVG.Use.class;
     %SVG.clipPath.extra.content; )*, ( %SVG.Animation.class; %SVG.Use.class;
     %SVG.Shape.class; )?, ( %SVG.Animation.class; %SVG.Use.class;
     %SVG.clipPath.extra.content; )*))"
>
<!ELEMENT %SVG.clipPath.qname; %SVG.clipPath.content; >
<!-- end of SVG.clipPath.element -->]]>

<!ENTITY % SVG.clipPath.attlist "INCLUDE" >
<![%SVG.clipPath.attlist;[
<!ATTLIST %SVG.clipPath.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Text.attrib;
  %SVG.TextContent.attrib;
  %SVG.Font.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  transform %TransformList.datatype; #IMPLIED
  clipPathUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
>
<!-- end of SVG.clipPath.attlist -->]]>

<!-- end of svg-basic-clip.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-basic-clip.mod
```

No license file was found, but licenses were detected in source scan.

```
<!--
   File isolat2.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

   Entity names in this file are derived from files carrying the
   following notice:

   (C) International Organization for Standardization 1986
   Permission to copy in any form is granted for use with
   conforming SGML systems and applications as defined in
   ISO 8879, provided this notice is included in all copies.

-->


<!--

     Public identifier: -//W3C//ENTITIES Added Latin 2//EN//XML
     System identifier: http://www.w3.org/2003/entities/2007/isolat2.ent

   The public identifier should always be used verbatim.
   The system identifier may be changed to suit local requirements.

   Typical invocation:

     <!ENTITY % isolat2 PUBLIC
       "-//W3C//ENTITIES Added Latin 2//EN//XML"
       "http://www.w3.org/2003/entities/2007/isolat2.ent"
     >
     %isolat2;

-->

<!ENTITY abreve          "&#x00103;" ><!--LATIN SMALL LETTER A WITH BREVE -->
```

```
<!ENTITY Abreve      "&#x00102;" ><!--LATIN CAPITAL LETTER A WITH BREVE -->
<!ENTITY amacr       "&#x00101;" ><!--LATIN SMALL LETTER A WITH MACRON -->
<!ENTITY Amacr       "&#x00100;" ><!--LATIN CAPITAL LETTER A WITH MACRON -->
<!ENTITY aogon       "&#x00105;" ><!--LATIN SMALL LETTER A WITH OGONEK -->
<!ENTITY Aogon       "&#x00104;" ><!--LATIN CAPITAL LETTER A WITH OGONEK -->
<!ENTITY cacute      "&#x00107;" ><!--LATIN SMALL LETTER C WITH ACUTE -->
<!ENTITY Cacute      "&#x00106;" ><!--LATIN CAPITAL LETTER C WITH ACUTE -->
<!ENTITY ccaron      "&#x0010D;" ><!--LATIN SMALL LETTER C WITH CARON -->
<!ENTITY Ccaron      "&#x0010C;" ><!--LATIN CAPITAL LETTER C WITH CARON -->
<!ENTITY ccirc       "&#x00109;" ><!--LATIN SMALL LETTER C WITH CIRCUMFLEX -->
<!ENTITY Ccirc       "&#x00108;" ><!--LATIN CAPITAL LETTER C WITH CIRCUMFLEX -->
<!ENTITY cdot        "&#x0010B;" ><!--LATIN SMALL LETTER C WITH DOT ABOVE -->
<!ENTITY Cdot        "&#x0010A;" ><!--LATIN CAPITAL LETTER C WITH DOT ABOVE -->
<!ENTITY dcaron      "&#x0010F;" ><!--LATIN SMALL LETTER D WITH CARON -->
<!ENTITY Dcaron      "&#x0010E;" ><!--LATIN CAPITAL LETTER D WITH CARON -->
<!ENTITY dstrok      "&#x00111;" ><!--LATIN SMALL LETTER D WITH STROKE -->
<!ENTITY Dstrok      "&#x00110;" ><!--LATIN CAPITAL LETTER D WITH STROKE -->
<!ENTITY ecaron      "&#x0011B;" ><!--LATIN SMALL LETTER E WITH CARON -->
<!ENTITY Ecaron      "&#x0011A;" ><!--LATIN CAPITAL LETTER E WITH CARON -->
<!ENTITY edot        "&#x00117;" ><!--LATIN SMALL LETTER E WITH DOT ABOVE -->
<!ENTITY Edot        "&#x00116;" ><!--LATIN CAPITAL LETTER E WITH DOT ABOVE -->
<!ENTITY emacr       "&#x00113;" ><!--LATIN SMALL LETTER E WITH MACRON -->
<!ENTITY Emacr       "&#x00112;" ><!--LATIN CAPITAL LETTER E WITH MACRON -->
<!ENTITY eng         "&#x0014B;" ><!--LATIN SMALL LETTER ENG -->
<!ENTITY ENG         "&#x0014A;" ><!--LATIN CAPITAL LETTER ENG -->
<!ENTITY eogon       "&#x00119;" ><!--LATIN SMALL LETTER E WITH OGONEK -->
<!ENTITY Eogon       "&#x00118;" ><!--LATIN CAPITAL LETTER E WITH OGONEK -->
<!ENTITY gacute      "&#x001F5;" ><!--LATIN SMALL LETTER G WITH ACUTE -->
<!ENTITY gbreve      "&#x0011F;" ><!--LATIN SMALL LETTER G WITH BREVE -->
<!ENTITY Gbreve      "&#x0011E;" ><!--LATIN CAPITAL LETTER G WITH BREVE -->
<!ENTITY Gcedil      "&#x00122;" ><!--LATIN CAPITAL LETTER G WITH CEDILLA -->
<!ENTITY gcirc       "&#x0011D;" ><!--LATIN SMALL LETTER G WITH CIRCUMFLEX -->
<!ENTITY Gcirc       "&#x0011C;" ><!--LATIN CAPITAL LETTER G WITH CIRCUMFLEX -->
<!ENTITY gdot        "&#x00121;" ><!--LATIN SMALL LETTER G WITH DOT ABOVE -->
<!ENTITY Gdot        "&#x00120;" ><!--LATIN CAPITAL LETTER G WITH DOT ABOVE -->
<!ENTITY hcirc       "&#x00125;" ><!--LATIN SMALL LETTER H WITH CIRCUMFLEX -->
<!ENTITY Hcirc       "&#x00124;" ><!--LATIN CAPITAL LETTER H WITH CIRCUMFLEX -->
<!ENTITY hstrok      "&#x00127;" ><!--LATIN SMALL LETTER H WITH STROKE -->
<!ENTITY Hstrok      "&#x00126;" ><!--LATIN CAPITAL LETTER H WITH STROKE -->
<!ENTITY Idot        "&#x00130;" ><!--LATIN CAPITAL LETTER I WITH DOT ABOVE -->
<!ENTITY ijlig       "&#x00133;" ><!--LATIN SMALL LIGATURE IJ -->
<!ENTITY IJlig       "&#x00132;" ><!--LATIN CAPITAL LIGATURE IJ -->
<!ENTITY imacr       "&#x0012B;" ><!--LATIN SMALL LETTER I WITH MACRON -->
<!ENTITY Imacr       "&#x0012A;" ><!--LATIN CAPITAL LETTER I WITH MACRON -->
<!ENTITY inodot      "&#x00131;" ><!--LATIN SMALL LETTER DOTLESS I -->
<!ENTITY iogon       "&#x0012F;" ><!--LATIN SMALL LETTER I WITH OGONEK -->
<!ENTITY Iogon       "&#x0012E;" ><!--LATIN CAPITAL LETTER I WITH OGONEK -->
<!ENTITY itilde      "&#x00129;" ><!--LATIN SMALL LETTER I WITH TILDE -->
```

```
<!ENTITY Itilde        "&#x00128;" ><!--LATIN CAPITAL LETTER I WITH TILDE -->
<!ENTITY jcirc         "&#x00135;" ><!--LATIN SMALL LETTER J WITH CIRCUMFLEX -->
<!ENTITY Jcirc         "&#x00134;" ><!--LATIN CAPITAL LETTER J WITH CIRCUMFLEX -->
<!ENTITY kcedil         "&#x00137;" ><!--LATIN SMALL LETTER K WITH CEDILLA -->
<!ENTITY Kcedil         "&#x00136;" ><!--LATIN CAPITAL LETTER K WITH CEDILLA -->
<!ENTITY kgreen         "&#x00138;" ><!--LATIN SMALL LETTER KRA -->
<!ENTITY lacute        "&#x0013A;" ><!--LATIN SMALL LETTER L WITH ACUTE -->
<!ENTITY Lacute         "&#x00139;" ><!--LATIN CAPITAL LETTER L WITH ACUTE -->
<!ENTITY lcaron        "&#x0013E;" ><!--LATIN SMALL LETTER L WITH CARON -->
<!ENTITY Lcaron         "&#x0013D;" ><!--LATIN CAPITAL LETTER L WITH CARON -->
<!ENTITY lcedil        "&#x0013C;" ><!--LATIN SMALL LETTER L WITH CEDILLA -->
<!ENTITY Lcedil         "&#x0013B;" ><!--LATIN CAPITAL LETTER L WITH CEDILLA -->
<!ENTITY lmidot         "&#x00140;" ><!--LATIN SMALL LETTER L WITH MIDDLE DOT -->
<!ENTITY Lmidot         "&#x0013F;" ><!--LATIN CAPITAL LETTER L WITH MIDDLE DOT -->
<!ENTITY lstrok        "&#x00142;" ><!--LATIN SMALL LETTER L WITH STROKE -->
<!ENTITY Lstrok         "&#x00141;" ><!--LATIN CAPITAL LETTER L WITH STROKE -->
<!ENTITY nacute         "&#x00144;" ><!--LATIN SMALL LETTER N WITH ACUTE -->
<!ENTITY Nacute         "&#x00143;" ><!--LATIN CAPITAL LETTER N WITH ACUTE -->
<!ENTITY napos          "&#x00149;" ><!--LATIN SMALL LETTER N PRECEDED BY APOSTROPHE -->
<!ENTITY ncaron         "&#x00148;" ><!--LATIN SMALL LETTER N WITH CARON -->
<!ENTITY Ncaron         "&#x00147;" ><!--LATIN CAPITAL LETTER N WITH CARON -->
<!ENTITY ncedil        "&#x00146;" ><!--LATIN SMALL LETTER N WITH CEDILLA -->
<!ENTITY Ncedil         "&#x00145;" ><!--LATIN CAPITAL LETTER N WITH CEDILLA -->
<!ENTITY odblac         "&#x00151;" ><!--LATIN SMALL LETTER O WITH DOUBLE ACUTE -->
<!ENTITY Odblac         "&#x00150;" ><!--LATIN CAPITAL LETTER O WITH DOUBLE ACUTE -->
<!ENTITY oelig         "&#x00153;" ><!--LATIN SMALL LIGATURE OE -->
<!ENTITY OElig          "&#x00152;" ><!--LATIN CAPITAL LIGATURE OE -->
<!ENTITY omacr          "&#x0014D;" ><!--LATIN SMALL LETTER O WITH MACRON -->
<!ENTITY Omacr          "&#x0014C;" ><!--LATIN CAPITAL LETTER O WITH MACRON -->
<!ENTITY racute        "&#x00155;" ><!--LATIN SMALL LETTER R WITH ACUTE -->
<!ENTITY Racute         "&#x00154;" ><!--LATIN CAPITAL LETTER R WITH ACUTE -->
<!ENTITY rcaron        "&#x00159;" ><!--LATIN SMALL LETTER R WITH CARON -->
<!ENTITY Rcaron         "&#x00158;" ><!--LATIN CAPITAL LETTER R WITH CARON -->
<!ENTITY rcedil        "&#x00157;" ><!--LATIN SMALL LETTER R WITH CEDILLA -->
<!ENTITY Rcedil         "&#x00156;" ><!--LATIN CAPITAL LETTER R WITH CEDILLA -->
<!ENTITY sacute        "&#x0015B;" ><!--LATIN SMALL LETTER S WITH ACUTE -->
<!ENTITY Sacute         "&#x0015A;" ><!--LATIN CAPITAL LETTER S WITH ACUTE -->
<!ENTITY scaron         "&#x00161;" ><!--LATIN SMALL LETTER S WITH CARON -->
<!ENTITY Scaron         "&#x00160;" ><!--LATIN CAPITAL LETTER S WITH CARON -->
<!ENTITY scedil        "&#x0015F;" ><!--LATIN SMALL LETTER S WITH CEDILLA -->
<!ENTITY Scedil         "&#x0015E;" ><!--LATIN CAPITAL LETTER S WITH CEDILLA -->
<!ENTITY scirc         "&#x0015D;" ><!--LATIN SMALL LETTER S WITH CIRCUMFLEX -->
<!ENTITY Scirc         "&#x0015C;" ><!--LATIN CAPITAL LETTER S WITH CIRCUMFLEX -->
<!ENTITY tcaron         "&#x00165;" ><!--LATIN SMALL LETTER T WITH CARON -->
<!ENTITY Tcaron         "&#x00164;" ><!--LATIN CAPITAL LETTER T WITH CARON -->
<!ENTITY tcedil        "&#x00163;" ><!--LATIN SMALL LETTER T WITH CEDILLA -->
<!ENTITY Tcedil         "&#x00162;" ><!--LATIN CAPITAL LETTER T WITH CEDILLA -->
<!ENTITY tstrok        "&#x00167;" ><!--LATIN SMALL LETTER T WITH STROKE -->
```

```
<!ENTITY Tstrok         "&#x00166;" ><!--LATIN CAPITAL LETTER T WITH STROKE -->
<!ENTITY ubreve         "&#x0016D;" ><!--LATIN SMALL LETTER U WITH BREVE -->
<!ENTITY Ubreve         "&#x0016C;" ><!--LATIN CAPITAL LETTER U WITH BREVE -->
<!ENTITY udblac         "&#x00171;" ><!--LATIN SMALL LETTER U WITH DOUBLE ACUTE -->
<!ENTITY Udblac         "&#x00170;" ><!--LATIN CAPITAL LETTER U WITH DOUBLE ACUTE -->
<!ENTITY umacr          "&#x0016B;" ><!--LATIN SMALL LETTER U WITH MACRON -->
<!ENTITY Umacr          "&#x0016A;" ><!--LATIN CAPITAL LETTER U WITH MACRON -->
<!ENTITY uogon          "&#x00173;" ><!--LATIN SMALL LETTER U WITH OGONEK -->
<!ENTITY Uogon          "&#x00172;" ><!--LATIN CAPITAL LETTER U WITH OGONEK -->
<!ENTITY uring          "&#x0016F;" ><!--LATIN SMALL LETTER U WITH RING ABOVE -->
<!ENTITY Uring          "&#x0016E;" ><!--LATIN CAPITAL LETTER U WITH RING ABOVE -->
<!ENTITY utilde         "&#x00169;" ><!--LATIN SMALL LETTER U WITH TILDE -->
<!ENTITY Utilde         "&#x00168;" ><!--LATIN CAPITAL LETTER U WITH TILDE -->
<!ENTITY wcirc          "&#x00175;" ><!--LATIN SMALL LETTER W WITH CIRCUMFLEX -->
<!ENTITY Wcirc          "&#x00174;" ><!--LATIN CAPITAL LETTER W WITH CIRCUMFLEX -->
<!ENTITY ycirc          "&#x00177;" ><!--LATIN SMALL LETTER Y WITH CIRCUMFLEX -->
<!ENTITY Ycirc          "&#x00176;" ><!--LATIN CAPITAL LETTER Y WITH CIRCUMFLEX -->
<!ENTITY Yuml           "&#x00178;" ><!--LATIN CAPITAL LETTER Y WITH DIAERESIS -->
<!ENTITY zacute         "&#x0017A;" ><!--LATIN SMALL LETTER Z WITH ACUTE -->
<!ENTITY Zacute         "&#x00179;" ><!--LATIN CAPITAL LETTER Z WITH ACUTE -->
<!ENTITY zcaron         "&#x0017E;" ><!--LATIN SMALL LETTER Z WITH CARON -->
<!ENTITY Zcaron         "&#x0017D;" ><!--LATIN CAPITAL LETTER Z WITH CARON -->
<!ENTITY zdot           "&#x0017C;" ><!--LATIN SMALL LETTER Z WITH DOT ABOVE -->
<!ENTITY Zdot           "&#x0017B;" ><!--LATIN CAPITAL LETTER Z WITH DOT ABOVE -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isolat2.ent
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Structure Module  ..................................... -->
<!-- file: xhtml-struct-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-struct-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Document Structure 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-struct-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Document Structure
```

```
      title, head, body, html

   The Structure Module defines the major structural elements and
   their attributes.

   Note that the content model of the head element type is redeclared
   when the Base Module is included in the DTD.

   The parameter entity containing the XML namespace URI value used
   for XHTML is '%XHTML.xmlns;', defined in the Qualified Names module.
-->

<!-- title: Document Title ..... -->

<!-- The title element is not considered part of the flow of text.
   It should be displayed, for example as the page header or
   window title. Exactly one title is required per document.
-->

<!ENTITY % title.element  "INCLUDE" >
<![%title.element;[
<!ENTITY % title.content  "( #PCDATA )" >
<!ENTITY % title.qname  "title" >
<!ELEMENT %title.qname;  %title.content; >
<!-- end of title.element -->]]>

<!ENTITY % title.attlist  "INCLUDE" >
<![%title.attlist;[
<!ATTLIST %title.qname;
    %XHTML.xmlns.attrib;
    %I18n.attrib;
>
<!-- end of title.attlist -->]]>

<!-- head: Document Head ..... -->

<!ENTITY % head.element  "INCLUDE" >
<![%head.element;[
<!ENTITY % head.content
  "( %HeadOpts.mix;, %title.qname;, %HeadOpts.mix; )"
>
<!ENTITY % head.qname  "head" >
<!ELEMENT %head.qname;  %head.content; >
<!-- end of head.element -->]]>

<!ENTITY % head.attlist  "INCLUDE" >
<![%head.attlist;[
```

```
<!-- reserved for future use with document profiles
-->
<!ENTITY % profile.attrib
   "profile      %URI.datatype;          '%XHTML.profile;'"
>

<!ATTLIST %head.qname;
    %XHTML.xmlns.attrib;
    %I18n.attrib;
    %profile.attrib;
    %id.attrib;
>
<!-- end of head.attlist -->]]>

<!-- body: Document Body ..... -->

<!ENTITY % body.element  "INCLUDE" >
<![%body.element;[
<!ENTITY % body.content
   "( %Block.mix; )*"
>
<!ENTITY % body.qname  "body" >
<!ELEMENT %body.qname;  %body.content; >
<!-- end of body.element -->]]>

<!ENTITY % body.attlist  "INCLUDE" >
<![%body.attlist;[
<!ATTLIST %body.qname;
    %Common.attrib;
>
<!-- end of body.attlist -->]]>

<!-- html: XHTML Document Element ..... -->

<!ENTITY % html.element  "INCLUDE" >
<![%html.element;[
<!ENTITY % html.content  "( %head.qname;, %body.qname; )" >
<!ENTITY % html.qname  "html" >
<!ELEMENT %html.qname;  %html.content; >
<!-- end of html.element -->]]>

<![%XHTML.xsi.attrs;[
<!-- define a parameter for the XSI schemaLocation attribute -->
<!ENTITY % XSI.schemaLocation.attrib
   "%XSI.pfx;schemaLocation  %URIs.datatype;    #IMPLIED"
>
]]>
<!ENTITY % XSI.schemaLocation.attrib "">
```

```
<!ENTITY % html.attlist  "INCLUDE" >
<![%html.attlist;[
<!-- version attribute value defined in driver
-->
<!ENTITY % XHTML.version.attrib
   "version      %FPI.datatype;          #FIXED '%XHTML.version;'"
>

<!-- see the Qualified Names module for information
   on how to extend XHTML using XML namespaces
-->
<!ATTLIST %html.qname;
    %XHTML.xmlns.attrib;
    %XSI.schemaLocation.attrib;
    %XHTML.version.attrib;
    %I18n.attrib;
    %id.attrib;
>
<!-- end of html.attlist -->]]>

<!-- end of xhtml-struct-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-struct-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Intrinsic Events Module  ..................................... -->
<!-- file: xhtml-events-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-events-1.mod,v 4.1 2001/04/10 09:42:30 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Intrinsic Events 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-events-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Intrinsic Event Attributes

   These are the event attributes defined in HTML 4,
```

```
   Section 18.2.3 "Intrinsic Events". This module must be
   instantiated prior to the Attributes Module but after
   the Datatype Module in the Modular Framework module.

  "Note: Authors of HTML documents are advised that changes
   are likely to occur in the realm of intrinsic events
   (e.g., how scripts are bound to events). Research in
   this realm is carried on by members of the W3C Document
   Object Model Working Group (see the W3C Web site at
   http://www.w3.org/ for more information)."
-->
<!-- NOTE: Because the ATTLIST declarations in this module occur
   before their respective ELEMENT declarations in other
   modules, there may be a dependency on this module that
   should be considered if any of the parameter entities used
   for element type names (eg., %a.qname;) are redeclared.
-->

<!ENTITY % Events.attrib
   "onclick      %Script.datatype;       #IMPLIED
    ondblclick   %Script.datatype;       #IMPLIED
    onmousedown  %Script.datatype;       #IMPLIED
    onmouseup    %Script.datatype;       #IMPLIED
    onmouseover  %Script.datatype;       #IMPLIED
    onmousemove  %Script.datatype;       #IMPLIED
    onmouseout   %Script.datatype;       #IMPLIED
    onkeypress   %Script.datatype;       #IMPLIED
    onkeydown    %Script.datatype;       #IMPLIED
    onkeyup      %Script.datatype;       #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.events.attrib
   "%XHTML.prefix;:onclick      %Script.datatype;       #IMPLIED
    %XHTML.prefix;:ondblclick   %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onmousedown  %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onmouseup    %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onmouseover  %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onmousemove  %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onmouseout   %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onkeypress   %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onkeydown    %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onkeyup      %Script.datatype;       #IMPLIED"
>
]]>

<!-- additional attributes on anchor element
-->
```

```
<!ATTLIST %a.qname;
   onfocus      %Script.datatype;        #IMPLIED
   onblur       %Script.datatype;        #IMPLIED
>

<!-- additional attributes on form element
-->
<!ATTLIST %form.qname;
    onsubmit     %Script.datatype;       #IMPLIED
    onreset      %Script.datatype;       #IMPLIED
>

<!-- additional attributes on label element
-->
<!ATTLIST %label.qname;
    onfocus      %Script.datatype;       #IMPLIED
    onblur       %Script.datatype;       #IMPLIED
>

<!-- additional attributes on input element
-->
<!ATTLIST %input.qname;
    onfocus      %Script.datatype;       #IMPLIED
    onblur       %Script.datatype;       #IMPLIED
    onselect     %Script.datatype;       #IMPLIED
    onchange     %Script.datatype;       #IMPLIED
>

<!-- additional attributes on select element
-->
<!ATTLIST %select.qname;
    onfocus      %Script.datatype;       #IMPLIED
    onblur       %Script.datatype;       #IMPLIED
    onchange     %Script.datatype;       #IMPLIED
>

<!-- additional attributes on textarea element
-->
<!ATTLIST %textarea.qname;
    onfocus      %Script.datatype;       #IMPLIED
    onblur       %Script.datatype;       #IMPLIED
    onselect     %Script.datatype;       #IMPLIED
    onchange     %Script.datatype;       #IMPLIED
>

<!-- additional attributes on button element
-->
<!ATTLIST %button.qname;
```

```
    onfocus      %Script.datatype;       #IMPLIED
    onblur       %Script.datatype;       #IMPLIED
>

<!-- additional attributes on body element
-->
<!ATTLIST %body.qname;
    onload       %Script.datatype;       #IMPLIED
    onunload     %Script.datatype;       #IMPLIED
>

<!-- additional attributes on area element
-->
<!ATTLIST %area.qname;
    onfocus      %Script.datatype;       #IMPLIED
    onblur       %Script.datatype;       #IMPLIED
>

<!-- end of xhtml-events-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-events-1.mod
No license file was found, but licenses were detected in source scan.

dimensional graphics in XML.

   The Scalable Vector Graphics (SVG)
   Copyright 2001, 2002 World Wide Web Consortium
     (Massachusetts Institute of Technology, Institut National de
      Recherche en Informatique et en Automatique, Keio University).
      All Rights Reserved.

   Permission to use, copy, modify and distribute the SVG DTD and its
   accompanying documentation for any purpose and without fee is hereby
   granted in perpetuity, provided that the above copyright notice and
   this paragraph appear in all copies.  The copyright holders make no
   representation about the suitability of the DTD for any purpose.

   It is provided "as is" without expressed or implied warranty.

     Author:   Jun Fujisawa <fujisawa.jun@canon.co.jp>
     Revision: $Id: svg11.dtd,v 1.2 2002/04/20 18:07:43 fujisawa Exp $
dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg

Found in path(s):
```

* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg11-flat.dtd
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg11-flat-20030114.dtd
No license file was found, but licenses were detected in source scan.

```
<!-- ..... -->
<!-- XHTML Name Identifier Module  ..... -->
<!-- file: xhtml-nameident-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-nameident-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Name Identifier 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-nameident-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Name Identifier

     'name' attribute on form, img, a, map, applet, frame, iframe

   This module declares the 'name' attribute on element types when
   it is used as a node identifier for legacy linking and scripting
   support. This does not include those instances when 'name' is used
   as a container for form control, property or metainformation names.

   This module should be instantiated following all modules it modifies.
-->

<!ENTITY % form.attlist  "IGNORE" >
<![%form.attlist;[
<!ATTLIST %form.qname;
    name        CDATA                #IMPLIED
>
<!-- end of form.attlist -->]]>

<!ENTITY % img.attlist  "IGNORE" >
<![%img.attlist;[
<!ATTLIST %img.qname;
    name        CDATA                #IMPLIED
>
<!-- end of img.attlist -->]]>
```

```
<!ENTITY % a.attlist  "IGNORE" >
<![%a.attlist;[
<!ATTLIST %a.qname;
    name        CDATA                #IMPLIED
>
<!-- end of a.attlist -->]]>

<!ENTITY % map.attlist  "IGNORE" >
<![%map.attlist;[
<!ATTLIST %map.qname;
    name        CDATA                #IMPLIED
>
<!-- end of map.attlist -->]]>

<!ENTITY % applet.attlist  "IGNORE" >
<![%applet.attlist;[
<!ATTLIST %applet.qname;
    name        CDATA                #IMPLIED
>
<!-- end of applet.attlist -->]]>

<!ENTITY % frame.attlist  "IGNORE" >
<![%frame.attlist;[
<!ATTLIST %frame.qname;
    name        CDATA                #IMPLIED
>
<!-- end of frame.attlist -->]]>

<!ENTITY % iframe.attlist  "IGNORE" >
<![%iframe.attlist;[
<!ATTLIST %iframe.qname;
    name        CDATA                #IMPLIED
>
<!-- end of iframe.attlist -->]]>

<!-- end of xhtml-nameident.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-nameident-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Basic 1.1 Document Model Module  ..................................... -->
<!-- file: xhtml-basic11-model-1.mod

   This is XHTML Basic, a proper subset of XHTML.
```

```
   Copyright 1998-2007 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-basic11-model-1.mod,v 1.4 2008/03/28 13:07:30 smccarro Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Basic 1.1 Document Model 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-basic11-model-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- XHTML Basic Document Model

   This module describes the groupings of elements that make up
   common content models for XHTML elements.
-->

<!-- Optional Elements in head  ..... -->

<!ENTITY % HeadOpts.mix
   "( %script.qname; | %style.qname; | %meta.qname;
    | %link.qname; | %object.qname; )*" >

<!-- script and noscript are used to contain scripts
   and alternative content
-->
<!ENTITY % Script.class "| %script.qname; | %noscript.qname;" >

<!-- Miscellaneous Elements  ..... -->

<!ENTITY % Misc.extra "" >

<!-- These elements are neither block nor inline, and can
   essentially be used anywhere in the document body.
-->
<!ENTITY % Misc.class
   "%Script.class;
    %Misc.extra;"
>

<!-- Inline Elements  ..... -->

<!ENTITY % InlStruct.class "%br.qname; | %span.qname;" >

<!ENTITY % InlPhras.class
   "| %em.qname; | %strong.qname; | %dfn.qname; | %code.qname;
    | %samp.qname; | %kbd.qname; | %var.qname; | %cite.qname;
```

```
    | %abbr.qname; | %acronym.qname; | %q.qname;" >

<!ENTITY % InlPres.class
   "| %tt.qname; | %i.qname; | %b.qname; | %big.qname;
    | %small.qname; | %sub.qname; | %sup.qname;" >

<!ENTITY % I18n.class "" >

<!ENTITY % Anchor.class "| %a.qname;" >

<!ENTITY % InlSpecial.class "| %img.qname; | %object.qname;" >

<!ENTITY % InlForm.class
   "| %input.qname; | %select.qname; | %textarea.qname;
    | %label.qname; | %button.qname;" >

<!ENTITY % Inline.extra "" >

<!ENTITY % Inline.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %Anchor.class;
    %InlSpecial.class;
    %InlForm.class;
    %Inline.extra;"
>

<!ENTITY % InlNoAnchor.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %InlSpecial.class;
    %InlForm.class;
    %Inline.extra;"
>

<!ENTITY % InlNoAnchor.mix
   "%InlNoAnchor.class;
    %Misc.class;"
>

<!ENTITY % Inline.mix
   "%Inline.class;
    %Misc.class;"
>

<!-- Block Elements  ..... -->
```

```
<!ENTITY % Heading.class
   "%h1.qname; | %h2.qname; | %h3.qname;
    | %h4.qname; | %h5.qname; | %h6.qname;"
>
<!ENTITY % List.class  "%ul.qname; | %ol.qname; | %dl.qname;" >

<!ENTITY % Table.class "| %table.qname;" >

<!ENTITY % Form.class  "| %form.qname;" >

<!ENTITY % Fieldset.class  "| %fieldset.qname;" >

<!ENTITY % BlkStruct.class "%p.qname; | %div.qname;" >

<!ENTITY % BlkPhras.class
   "| %pre.qname; | %blockquote.qname; | %address.qname;"
>

<!ENTITY % BlkPres.class "| %hr.qname;" >

<!ENTITY % BlkSpecial.class
   "%Table.class;
    %Form.class;
    %Fieldset.class;"
>

<!ENTITY % Block.extra "" >

<!ENTITY % Block.class
   "%BlkStruct.class;
    %BlkPhras.class;
    %BlkPres.class;
    %BlkSpecial.class;
    %Block.extra;"
>

<!ENTITY % Block.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    %Misc.class;"
>

<!-- All Content Elements  ..... -->

<!-- declares all content except tables
-->
```

```
<!ENTITY % FlowNoTable.mix
   "%Heading.class;
    | %List.class;
    | %BlkStruct.class;
    %BlkPhras.class;
    %Form.class;
    %Block.extra;
    | %Inline.class;
    %Misc.class;"
>

<!ENTITY % Flow.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    | %Inline.class;
    %Misc.class;"
>

<!-- end of xhtml-basic11-model-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-basic11-model-1.mod
No license file was found, but licenses were detected in source scan.

SVG 1.1 Basic DTD

This is SVG Basic, a proper subset of SVG.

The Scalable Vector Graphics (SVG)
Copyright 2001, 2002 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the SVG DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11
dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.

```
   Revision: $Id: svg

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg11-basic-flat.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Basic Attribute Collection Module ............................ -->
<!-- file: svg11-basic-attribs.mod

   This is SVG Basic, a proper subset of SVG.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg11-basic-attribs.mod,v 1.5 2002/11/14 15:11:03 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Basic Attribute Collection//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11-basic-attribs.mod"

   ..................................................................... -->

<!-- SVG 1.1 Basic Attribute Collection

   This module defines the set of common attributes that can be present
   on many SVG elements.
-->

<!-- module: svg-conditional.mod ....................... -->

<!ENTITY % ExtensionList.datatype "CDATA" >
<!ENTITY % FeatureList.datatype "CDATA" >

<!ENTITY % SVG.Conditional.extra.attrib "" >
<!ENTITY % SVG.Conditional.attrib
  "requiredFeatures %FeatureList.datatype; #IMPLIED
   requiredExtensions %ExtensionList.datatype; #IMPLIED
   systemLanguage %LanguageCodes.datatype; #IMPLIED
   %SVG.Conditional.extra.attrib;"
>

<!-- module: svg-style.mod ............................. -->

<!ENTITY % ClassList.datatype "CDATA" >
<!ENTITY % StyleSheet.datatype "CDATA" >

<!ENTITY % SVG.Style.extra.attrib "" >
<!ENTITY % SVG.Style.attrib
```

```
  "style %StyleSheet.datatype; #IMPLIED
   class %ClassList.datatype; #IMPLIED
   %SVG.Style.extra.attrib;"
>

<!-- module: svg-text.mod ..... -->

<!ENTITY % BaselineShiftValue.datatype "CDATA" >
<!ENTITY % FontFamilyValue.datatype "CDATA" >
<!ENTITY % FontSizeValue.datatype "CDATA" >
<!ENTITY % FontSizeAdjustValue.datatype "CDATA" >
<!ENTITY % GlyphOrientationHorizontalValue.datatype "CDATA" >
<!ENTITY % GlyphOrientationVerticalValue.datatype "CDATA" >
<!ENTITY % KerningValue.datatype "CDATA" >
<!ENTITY % SpacingValue.datatype "CDATA" >
<!ENTITY % TextDecorationValue.datatype "CDATA" >

<!ENTITY % SVG.Text.extra.attrib "" >
<!ENTITY % SVG.Text.attrib
  "writing-mode ( lr-tb | rl-tb | tb-rl | lr | rl | tb | inherit ) #IMPLIED
   %SVG.Text.extra.attrib;"
>

<!ENTITY % SVG.TextContent.extra.attrib "" >
<!ENTITY % SVG.TextContent.attrib
  "alignment-baseline ( auto | baseline | before-edge | text-before-edge |
                        middle | central | after-edge | text-after-edge |
                        ideographic | alphabetic | hanging | mathematical |
                        inherit ) #IMPLIED
   baseline-shift %BaselineShiftValue.datatype; #IMPLIED
   direction ( ltr | rtl | inherit ) #IMPLIED
   dominant-baseline ( auto | use-script | no-change | reset-size |
                       ideographic | alphabetic | hanging | mathematical |
                       central | middle | text-after-edge | text-before-edge |
                       inherit ) #IMPLIED
   glyph-orientation-horizontal %GlyphOrientationHorizontalValue.datatype;
                           #IMPLIED
   glyph-orientation-vertical %GlyphOrientationVerticalValue.datatype;
                           #IMPLIED
   kerning %KerningValue.datatype; #IMPLIED
   letter-spacing %SpacingValue.datatype; #IMPLIED
   text-anchor ( start | middle | end | inherit ) #IMPLIED
   text-decoration %TextDecorationValue.datatype; #IMPLIED
   unicode-bidi ( normal | embed | bidi-override | inherit ) #IMPLIED
   word-spacing %SpacingValue.datatype; #IMPLIED
   %SVG.TextContent.extra.attrib;"
>
```

```
<!ENTITY % SVG.Font.extra.attrib "" >
<!ENTITY % SVG.Font.attrib
  "font-family %FontFamilyValue.datatype; #IMPLIED
   font-size %FontSizeValue.datatype; #IMPLIED
   font-size-adjust %FontSizeAdjustValue.datatype; #IMPLIED
   font-stretch ( normal | wider | narrower | ultra-condensed |
                  extra-condensed | condensed | semi-condensed |
                  semi-expanded | expanded | extra-expanded |
                  ultra-expanded | inherit ) #IMPLIED
   font-style ( normal | italic | oblique | inherit ) #IMPLIED
   font-variant ( normal | small-caps | inherit ) #IMPLIED
   font-weight ( normal | bold | bolder | lighter | 100 | 200 | 300 | 400 |
                 500 | 600 | 700 | 800 | 900 | inherit ) #IMPLIED
   %SVG.Font.extra.attrib;"
>

<!-- module: svg-profile.mod ..... -->

<!ENTITY % SVG.ColorProfile.extra.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib
  "color-profile CDATA #IMPLIED
   %SVG.ColorProfile.extra.attrib;"
>

<!-- module: svg-gradient.mod ..... -->

<!ENTITY % NumberOrPercentage.datatype "CDATA" >

<!ENTITY % SVG.Gradient.extra.attrib "" >
<!ENTITY % SVG.Gradient.attrib
  "stop-color %SVGColor.datatype; #IMPLIED
   stop-opacity %OpacityValue.datatype; #IMPLIED
   %SVG.Gradient.extra.attrib;"
>

<!-- module: svg-basic-clip.mod ..... -->

<!ENTITY % ClipPathValue.datatype "CDATA" >

<!ENTITY % SVG.Clip.extra.attrib "" >
<!ENTITY % SVG.Clip.attrib
  "clip-path %ClipPathValue.datatype; #IMPLIED
   clip-rule %ClipFillRule.datatype; #IMPLIED
   %SVG.Clip.extra.attrib;"
>

<!-- module: svg-mask.mod ..... -->
```

```
<!ENTITY % MaskValue.datatype "CDATA" >

<!ENTITY % SVG.Mask.extra.attrib "" >
<!ENTITY % SVG.Mask.attrib
  "mask %MaskValue.datatype; #IMPLIED
   %SVG.Mask.extra.attrib;"
>

<!-- module: svg-basic-filter.mod ..... -->

<!ENTITY % FilterValue.datatype "CDATA" >
<!ENTITY % NumberOptionalNumber.datatype "CDATA" >

<!ENTITY % SVG.Filter.extra.attrib "" >
<!ENTITY % SVG.Filter.attrib
  "filter %FilterValue.datatype; #IMPLIED
   %SVG.Filter.extra.attrib;"
>

<!ENTITY % SVG.FilterColor.extra.attrib "" >
<!ENTITY % SVG.FilterColor.attrib
  "color-interpolation-filters ( auto | sRGB | linearRGB | inherit )
                          #IMPLIED
   %SVG.FilterColor.extra.attrib;"
>

<!-- end of svg11-basic-attribs.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg11-basic-attribs.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Common Attributes Module  ..... -->
<!-- file: xhtml-attribs-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-attribs-1.mod,v 1.1 2001/02/13 12:24:21 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Common Attributes 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-attribs-1.mod"

   Revisions:
   (none)
```

```
    ..... -->

<!-- Common Attributes

    This module declares many of the common attributes for the XHTML DTD.
    %NS.decl.attrib; is declared in the XHTML Qname module.
-->

<!ENTITY % id.attrib
    "id          ID                    #IMPLIED"
>

<!ENTITY % class.attrib
    "class       NMTOKENS              #IMPLIED"
>

<!ENTITY % title.attrib
    "title       %Text.datatype;       #IMPLIED"
>

<!ENTITY % Core.extra.attrib "" >

<!ENTITY % Core.attrib
    "%XHTML.xmlns.attrib;
     %id.attrib;
     %class.attrib;
     %title.attrib;
     %Core.extra.attrib;"
>

<!ENTITY % lang.attrib
    "xml:lang    %LanguageCode.datatype;  #IMPLIED"
>

<![%XHTML.bidi;[
<!ENTITY % dir.attrib
    "dir         ( ltr | rtl )         #IMPLIED"
>

<!ENTITY % I18n.attrib
    "%dir.attrib;
     %lang.attrib;"
>

]]>
<!ENTITY % I18n.attrib
    "%lang.attrib;"
>
```

```
<!ENTITY % Common.extra.attrib "" >

<!-- intrinsic event attributes declared previously
-->
<!ENTITY % Events.attrib "" >

<!ENTITY % Common.attrib
   "%Core.attrib;
    %I18n.attrib;
    %Events.attrib;
    %Common.extra.attrib;"
>

<!-- end of xhtml-attribs-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-attribs-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Qualified Name Module ....................................... -->
<!-- file: svg-qname.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-qname.mod,v 1.3 2002/11/03 15:54:14 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Qualified Name//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-qname.mod"

   ..................................................................... -->

<!-- Qualified Name

   This module is contained in two parts, labeled Section 'A' and 'B':

     Section A declares parameter entities to support namespace-
     qualified names, namespace declarations, and name prefixing
     for SVG and extensions.

     Section B declares parameter entities used to provide
     namespace-qualified names for all SVG element types:
-->
```

```
<!-- Section A: SVG XML Namespace Framework :::::::::::::::::::: -->

<!-- 1. Declare a %SVG.prefixed; conditional section keyword, used
     to activate namespace prefixing. The default value should
     inherit '%NS.prefixed;' from the DTD driver, so that unless
     overridden, the default behaviour follows the overall DTD
     prefixing scheme.
-->
<!ENTITY % NS.prefixed "IGNORE" >
<!ENTITY % SVG.prefixed "%NS.prefixed;" >

<!-- 2. Declare a parameter entity (eg., %SVG.xmlns;) containing
     the URI reference used to identify the SVG namespace:
-->
<!ENTITY % SVG.xmlns "http://www.w3.org/2000/svg" >
<!ENTITY % XLINK.xmlns "http://www.w3.org/1999/xlink" >

<!-- 3. Declare parameter entities (eg., %SVG.prefix;) containing
     the default namespace prefix string(s) to use when prefixing
     is enabled. This may be overridden in the DTD driver or the
     internal subset of an document instance. If no default prefix
     is desired, this may be declared as an empty string.
-->
<!ENTITY % SVG.prefix "" >
<!ENTITY % XLINK.prefix "xlink" >

<!-- 4. Declare parameter entities (eg., %SVG.pfx;) containing the
     colonized prefix(es) (eg., '%SVG.prefix;:') used when
     prefixing is active, an empty string when it is not.
-->
<![%SVG.prefixed;[
<!ENTITY % SVG.pfx "%SVG.prefix;:" >
]]>
<!ENTITY % SVG.pfx "" >
<!ENTITY % XLINK.pfx "%XLINK.prefix;:" >

<!-- 5. The parameter entity %SVG.xmlns.extra.attrib; may be
     redeclared to contain any non-SVG namespace declaration
     attributes for namespaces embedded in SVG. The default
     is an empty string.
-->
<!ENTITY % SVG.xmlns.extra.attrib "" >

<!-- Declare a parameter entity XLINK.xmlns.attrib containing
   the XML Namespace declarations for XLink.
-->
<!ENTITY % XLINK.xmlns.attrib
   "xmlns:%XLINK.prefix; %URI.datatype; #FIXED '%XLINK.xmlns;'"
```

```
>

<!-- Declare a parameter entity %NS.decl.attrib; containing
    all XML Namespace declarations used in the DTD, plus the
    xmlns declaration for SVG, its form dependent on whether
    prefixing is active.
-->
<![%SVG.prefixed;[
<!ENTITY % NS.decl.attrib
  "xmlns:%SVG.prefix; %URI.datatype; #FIXED '%SVG.xmlns;'
    %XLINK.xmlns.attrib;
    %SVG.xmlns.extra.attrib;"
>
]]>
<!ENTITY % NS.decl.attrib
  "%XLINK.xmlns.attrib;
    %SVG.xmlns.extra.attrib;"
>

<!-- Declare a parameter entity %SVG.xmlns.attrib; containing
    all XML namespace declaration attributes used by SVG,
    including a default xmlns attribute when prefixing is
    inactive.
-->
<![%SVG.prefixed;[
<!ENTITY % SVG.xmlns.attrib
    "%NS.decl.attrib;"
>
]]>
<!ENTITY % SVG.xmlns.attrib
    "xmlns %URI.datatype; #FIXED '%SVG.xmlns;'
     %XLINK.xmlns.attrib;"
>

<!-- Section B: SVG Qualified Names :::::::::::::::::::::::::::::: -->

<!-- 6. This section declares parameter entities used to provide
       namespace-qualified names for all SVG element types.
-->

<!-- module: svg-structure.mod ..... -->

<!ENTITY % SVG.svg.qname "%SVG.pfx;svg" >
<!ENTITY % SVG.g.qname "%SVG.pfx;g" >
<!ENTITY % SVG.defs.qname "%SVG.pfx;defs" >
<!ENTITY % SVG.desc.qname "%SVG.pfx;desc" >
<!ENTITY % SVG.title.qname "%SVG.pfx;title" >
<!ENTITY % SVG.metadata.qname "%SVG.pfx;metadata" >
```

```
<!ENTITY % SVG.symbol.qname "%SVG.pfx;symbol" >
<!ENTITY % SVG.use.qname "%SVG.pfx;use" >

<!-- module: svg-conditional.mod ..... -->

<!ENTITY % SVG.switch.qname "%SVG.pfx;switch" >

<!-- module: svg-image.mod ..... -->

<!ENTITY % SVG.image.qname "%SVG.pfx;image" >

<!-- module: svg-style.mod ..... -->

<!ENTITY % SVG.style.qname "%SVG.pfx;style" >

<!-- module: svg-shape.mod ..... -->

<!ENTITY % SVG.path.qname "%SVG.pfx;path" >
<!ENTITY % SVG.rect.qname "%SVG.pfx;rect" >
<!ENTITY % SVG.circle.qname "%SVG.pfx;circle" >
<!ENTITY % SVG.line.qname "%SVG.pfx;line" >
<!ENTITY % SVG.ellipse.qname "%SVG.pfx;ellipse" >
<!ENTITY % SVG.polyline.qname "%SVG.pfx;polyline" >
<!ENTITY % SVG.polygon.qname "%SVG.pfx;polygon" >

<!-- module: svg-text.mod ..... -->

<!ENTITY % SVG.text.qname "%SVG.pfx;text" >
<!ENTITY % SVG.tspan.qname "%SVG.pfx;tspan" >
<!ENTITY % SVG.tref.qname "%SVG.pfx;tref" >
<!ENTITY % SVG.textPath.qname "%SVG.pfx;textPath" >
<!ENTITY % SVG.altGlyph.qname "%SVG.pfx;altGlyph" >
<!ENTITY % SVG.altGlyphDef.qname "%SVG.pfx;altGlyphDef" >
<!ENTITY % SVG.altGlyphItem.qname "%SVG.pfx;altGlyphItem" >
<!ENTITY % SVG.glyphRef.qname "%SVG.pfx;glyphRef" >

<!-- module: svg-marker.mod ..... -->

<!ENTITY % SVG.marker.qname "%SVG.pfx;marker" >

<!-- module: svg-profile.mod ..... -->

<!ENTITY % SVG.color-profile.qname "%SVG.pfx;color-profile" >

<!-- module: svg-gradient.mod ..... -->

<!ENTITY % SVG.linearGradient.qname "%SVG.pfx;linearGradient" >
<!ENTITY % SVG.radialGradient.qname "%SVG.pfx;radialGradient" >
```

```
<!ENTITY % SVG.stop.qname "%SVG.pfx;stop" >

<!-- module: svg-pattern.mod ..... -->

<!ENTITY % SVG.pattern.qname "%SVG.pfx;pattern" >

<!-- module: svg-clip.mod ..... -->

<!ENTITY % SVG.clipPath.qname "%SVG.pfx;clipPath" >

<!-- module: svg-mask.mod ..... -->

<!ENTITY % SVG.mask.qname "%SVG.pfx;mask" >

<!-- module: svg-filter.mod ..... -->

<!ENTITY % SVG.filter.qname "%SVG.pfx;filter" >
<!ENTITY % SVG.feBlend.qname "%SVG.pfx;feBlend" >
<!ENTITY % SVG.feColorMatrix.qname "%SVG.pfx;feColorMatrix" >
<!ENTITY % SVG.feComponentTransfer.qname "%SVG.pfx;feComponentTransfer" >
<!ENTITY % SVG.feComposite.qname "%SVG.pfx;feComposite" >
<!ENTITY % SVG.feConvolveMatrix.qname "%SVG.pfx;feConvolveMatrix" >
<!ENTITY % SVG.feDiffuseLighting.qname "%SVG.pfx;feDiffuseLighting" >
<!ENTITY % SVG.feDisplacementMap.qname "%SVG.pfx;feDisplacementMap" >
<!ENTITY % SVG.feFlood.qname "%SVG.pfx;feFlood" >
<!ENTITY % SVG.feGaussianBlur.qname "%SVG.pfx;feGaussianBlur" >
<!ENTITY % SVG.feImage.qname "%SVG.pfx;feImage" >
<!ENTITY % SVG.feMerge.qname "%SVG.pfx;feMerge" >
<!ENTITY % SVG.feMergeNode.qname "%SVG.pfx;feMergeNode" >
<!ENTITY % SVG.feMorphology.qname "%SVG.pfx;feMorphology" >
<!ENTITY % SVG.feOffset.qname "%SVG.pfx;feOffset" >
<!ENTITY % SVG.feSpecularLighting.qname "%SVG.pfx;feSpecularLighting" >
<!ENTITY % SVG.feTile.qname "%SVG.pfx;feTile" >
<!ENTITY % SVG.feTurbulence.qname "%SVG.pfx;feTurbulence" >
<!ENTITY % SVG.feDistantLight.qname "%SVG.pfx;feDistantLight" >
<!ENTITY % SVG.fePointLight.qname "%SVG.pfx;fePointLight" >
<!ENTITY % SVG.feSpotLight.qname "%SVG.pfx;feSpotLight" >
<!ENTITY % SVG.feFuncR.qname "%SVG.pfx;feFuncR" >
<!ENTITY % SVG.feFuncG.qname "%SVG.pfx;feFuncG" >
<!ENTITY % SVG.feFuncB.qname "%SVG.pfx;feFuncB" >
<!ENTITY % SVG.feFuncA.qname "%SVG.pfx;feFuncA" >

<!-- module: svg-cursor.mod ..... -->

<!ENTITY % SVG.cursor.qname "%SVG.pfx;cursor" >

<!-- module: svg-hyperlink.mod ..... -->
```

```
<!ENTITY % SVG.a.qname "%SVG.pfx;a" >

<!-- module: svg-view.mod ..... -->

<!ENTITY % SVG.view.qname "%SVG.pfx;view" >

<!-- module: svg-script.mod ..... -->

<!ENTITY % SVG.script.qname "%SVG.pfx;script" >

<!-- module: svg-animation.mod ..... -->

<!ENTITY % SVG.animate.qname "%SVG.pfx;animate" >
<!ENTITY % SVG.set.qname "%SVG.pfx;set" >
<!ENTITY % SVG.animateMotion.qname "%SVG.pfx;animateMotion" >
<!ENTITY % SVG.animateColor.qname "%SVG.pfx;animateColor" >
<!ENTITY % SVG.animateTransform.qname "%SVG.pfx;animateTransform" >
<!ENTITY % SVG.mpath.qname "%SVG.pfx;mpath" >

<!-- module: svg-font.mod ..... -->

<!ENTITY % SVG.font.qname "%SVG.pfx;font" >
<!ENTITY % SVG.font-face.qname "%SVG.pfx;font-face" >
<!ENTITY % SVG.glyph.qname "%SVG.pfx;glyph" >
<!ENTITY % SVG.missing-glyph.qname "%SVG.pfx;missing-glyph" >
<!ENTITY % SVG.hkern.qname "%SVG.pfx;hkern" >
<!ENTITY % SVG.vkern.qname "%SVG.pfx;vkern" >
<!ENTITY % SVG.font-face-src.qname "%SVG.pfx;font-face-src" >
<!ENTITY % SVG.font-face-uri.qname "%SVG.pfx;font-face-uri" >
<!ENTITY % SVG.font-face-format.qname "%SVG.pfx;font-face-format" >
<!ENTITY % SVG.font-face-name.qname "%SVG.pfx;font-face-name" >
<!ENTITY % SVG.definition-src.qname "%SVG.pfx;definition-src" >

<!-- module: svg-extensibility.mod ..... -->

<!ENTITY % SVG.foreignObject.qname "%SVG.pfx;foreignObject" >

<!-- end of svg-qname.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-qname.mod
No license file was found, but licenses were detected in source scan.

2009 W3C® (MIT, ERCIM, Keio), All Rights
Reserved. W3C liability, trademark, document use and software
licensing rules apply.

Permission to use, copy, modify and distribute the MathML 2.0 DTD and
its accompanying documentation for any purpose and without fee is
hereby granted in perpetuity, provided that the above copyright notice
and this paragraph appear in all copies. The copyright holders make
no representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

This entity may be identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/mathml3.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Base Architecture Module ..... -->
<!-- file: xhtml-arch-1.mod

This is XHTML, a reformulation of HTML as a modular XML application.
Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
Revision: $Id: xhtml-arch-1.mod,v 1.4 2008/10/08 21:02:30 jules Exp $ SMI

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS XHTML Base Architecture 1.0//EN"
SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-arch-1.mod"

Revisions:
(none)
..... -->

<!-- This optional module includes declarations that enable XHTML to be used
as a base architecture according to the 'Architectural Forms Definition
Requirements' (Annex A.3, ISO/IEC 10744, 2nd edition). For more information
on use of architectural forms, see the HyTime web site at:

http://www.hytime.org/
-->

<?IS10744 ArcBase xhtml ?>

<!NOTATION xhtml PUBLIC "-//W3C//NOTATION AFDR ARCBASE XHTML 1.1//EN" >

<!-- Entity declaration for associated Architectural DTD
-->

```
<!ENTITY xhtml-arch.dtd
    PUBLIC "-//W3C//DTD XHTML Architecture 1.1//EN"
          "xhtml11-arch.dtd" >

<?IS10744:arch xhtml
  public-id      =  "-//W3C//NOTATION AFDR ARCBASE XHTML 1.1//EN"
  dtd-public-id   =  "-//W3C//DTD XHTML 1.1//EN"
  dtd-system-id   =  "xhtml11.dtd"
  doc-elem-form   =  "html"
  form-att        =  "html"
  renamer-att     =  "htnames"
  suppressor-att  =  "htsupp"
  data-ignore-att =  "htign"
  auto            =  "ArcAuto"
  options         =  "HtModReq HtModOpt"
  HtModReq        =  "Framework Text Hypertext Lists Structure"
  HtModOpt        =  "Standard"
?>

<!-- end of xhtml-arch-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-arch-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Intrinsic Events Module  ....................................... -->
<!-- file: xhtml-events-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-events-1.mod,v 1.4 2008/10/08 21:02:30 jules Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Intrinsic Events 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-events-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Intrinsic Event Attributes

   These are the event attributes defined in HTML 4,
   Section 18.2.3 "Intrinsic Events". This module must be
   instantiated prior to the Attributes Module but after
```

```
   the Datatype Module in the Modular Framework module.

  "Note: Authors of HTML documents are advised that changes
   are likely to occur in the realm of intrinsic events
   (e.g., how scripts are bound to events). Research in
   this realm is carried on by members of the W3C Document
   Object Model Working Group (see the W3C Web site at
   http://www.w3.org/ for more information)."
-->
<!-- NOTE: Because the ATTLIST declarations in this module occur
   before their respective ELEMENT declarations in other
   modules, there may be a dependency on this module that
   should be considered if any of the parameter entities used
   for element type names (eg., %a.qname;) are redeclared.
-->

<!ENTITY % Events.attrib
   "onclick      %Script.datatype;       #IMPLIED
    ondblclick   %Script.datatype;       #IMPLIED
    onmousedown  %Script.datatype;       #IMPLIED
    onmouseup    %Script.datatype;       #IMPLIED
    onmouseover  %Script.datatype;       #IMPLIED
    onmousemove  %Script.datatype;       #IMPLIED
    onmouseout   %Script.datatype;       #IMPLIED
    onkeypress   %Script.datatype;       #IMPLIED
    onkeydown    %Script.datatype;       #IMPLIED
    onkeyup      %Script.datatype;       #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.events.attrib
   "%XHTML.prefix;:onclick      %Script.datatype;       #IMPLIED
    %XHTML.prefix;:ondblclick   %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onmousedown  %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onmouseup    %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onmouseover  %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onmousemove  %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onmouseout   %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onkeypress   %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onkeydown    %Script.datatype;       #IMPLIED
    %XHTML.prefix;:onkeyup      %Script.datatype;       #IMPLIED"
>
]]>

<!-- additional attributes on anchor element
-->
<!ATTLIST %a.qname;
   onfocus      %Script.datatype;       #IMPLIED
```

```
   onblur      %Script.datatype;       #IMPLIED
>

<!-- additional attributes on form element
-->
<!ATTLIST %form.qname;
    onsubmit    %Script.datatype;       #IMPLIED
    onreset     %Script.datatype;       #IMPLIED
>

<!-- additional attributes on label element
-->
<!ATTLIST %label.qname;
    onfocus     %Script.datatype;       #IMPLIED
    onblur      %Script.datatype;       #IMPLIED
>

<!-- additional attributes on input element
-->
<!ATTLIST %input.qname;
    onfocus     %Script.datatype;       #IMPLIED
    onblur      %Script.datatype;       #IMPLIED
    onselect    %Script.datatype;       #IMPLIED
    onchange    %Script.datatype;       #IMPLIED
>

<!-- additional attributes on select element
-->
<!ATTLIST %select.qname;
    onfocus     %Script.datatype;       #IMPLIED
    onblur      %Script.datatype;       #IMPLIED
    onchange    %Script.datatype;       #IMPLIED
>

<!-- additional attributes on textarea element
-->
<!ATTLIST %textarea.qname;
    onfocus     %Script.datatype;       #IMPLIED
    onblur      %Script.datatype;       #IMPLIED
    onselect    %Script.datatype;       #IMPLIED
    onchange    %Script.datatype;       #IMPLIED
>

<!-- additional attributes on button element
-->
<!ATTLIST %button.qname;
    onfocus     %Script.datatype;       #IMPLIED
    onblur      %Script.datatype;       #IMPLIED
```

```
>

<!-- additional attributes on body element
-->
<!ATTLIST %body.qname;
    onload      %Script.datatype;       #IMPLIED
    onunload    %Script.datatype;       #IMPLIED
>

<!-- additional attributes on area element
-->
<!ATTLIST %area.qname;
    onfocus     %Script.datatype;       #IMPLIED
    onblur      %Script.datatype;       #IMPLIED
>

<!-- end of xhtml-events-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-events-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- XHTML Basic Table Module  ..... -->
<!-- file: xhtml-basic-table-1.mod

   This is XHTML Basic, a proper subset of XHTML.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-basic-table-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Basic Tables 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-basic-table-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Basic Tables

      table, caption, tr, th, td

   This table module declares elements and attributes defining
   a table model based fundamentally on features found in the
   widely-deployed HTML 3.2 table model.  While this module
   mimics the content model and table attributes of HTML 3.2
```

```
   tables, the element types declared herein also includes all
   HTML 4 common and most of the HTML 4 table attributes.
-->

<!-- declare qualified element type names:
-->
<!ENTITY % table.qname  "table" >
<!ENTITY % caption.qname  "caption" >
<!ENTITY % tr.qname  "tr" >
<!ENTITY % th.qname  "th" >
<!ENTITY % td.qname  "td" >

<!-- horizontal alignment attributes for cell contents
-->
<!ENTITY % CellHAlign.attrib
   "align       ( left
              | center
              | right )              #IMPLIED"
>

<!-- vertical alignment attributes for cell contents
-->
<!ENTITY % CellVAlign.attrib
   "valign      ( top
              | middle
              | bottom )             #IMPLIED"
>

<!-- scope is simpler than axes attribute for common tables
-->
<!ENTITY % scope.attrib
   "scope       ( row | col  )         #IMPLIED"
>

<!-- table: Table Element ..... -->

<!ENTITY % table.element  "INCLUDE" >
<![%table.element;[
<!ENTITY % table.content
   "( %caption.qname;?, %tr.qname;+ )"
>
<!ELEMENT %table.qname;  %table.content; >
<!-- end of table.element -->]]>

<!ENTITY % table.attlist  "INCLUDE" >
<![%table.attlist;[
<!ATTLIST %table.qname;
    %Common.attrib;
```

```
    summary      %Text.datatype;         #IMPLIED
    width        %Length.datatype;       #IMPLIED
>
<!-- end of table.attlist -->]]>

<!-- caption: Table Caption ..... -->

<!ENTITY % caption.element  "INCLUDE" >
<![%caption.element;[
<!ENTITY % caption.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ELEMENT %caption.qname;  %caption.content; >
<!-- end of caption.element -->]]>

<!ENTITY % caption.attlist  "INCLUDE" >
<![%caption.attlist;[
<!ATTLIST %caption.qname;
    %Common.attrib;
>
<!-- end of caption.attlist -->]]>

<!-- tr: Table Row ..... -->

<!ENTITY % tr.element  "INCLUDE" >
<![%tr.element;[
<!ENTITY % tr.content  "( %th.qname; | %td.qname; )+" >
<!ELEMENT %tr.qname;  %tr.content; >
<!-- end of tr.element -->]]>

<!ENTITY % tr.attlist  "INCLUDE" >
<![%tr.attlist;[
<!ATTLIST %tr.qname;
    %Common.attrib;
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of tr.attlist -->]]>

<!-- th: Table Header Cell ..... -->

<!-- th is for header cells, td for data,
   but for cells acting as both use td
-->

<!ENTITY % th.element  "INCLUDE" >
<![%th.element;[
<!ENTITY % th.content
```

```
   "( #PCDATA | %FlowNoTable.mix; )*"
>
<!ELEMENT %th.qname;  %th.content; >
<!-- end of th.element -->]]>

<!ENTITY % th.attlist  "INCLUDE" >
<![%th.attlist;[
<!ATTLIST %th.qname;
    %Common.attrib;
    abbr        %Text.datatype;         #IMPLIED
    axis        CDATA                   #IMPLIED
    headers     IDREFS                  #IMPLIED
    %scope.attrib;
    rowspan     %Number.datatype;       '1'
    colspan     %Number.datatype;       '1'
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of th.attlist -->]]>

<!-- td: Table Data Cell ..... -->

<!ENTITY % td.element  "INCLUDE" >
<![%td.element;[
<!ENTITY % td.content
   "( #PCDATA | %FlowNoTable.mix; )*"
>
<!ELEMENT %td.qname;  %td.content; >
<!-- end of td.element -->]]>

<!ENTITY % td.attlist  "INCLUDE" >
<![%td.attlist;[
<!ATTLIST %td.qname;
    %Common.attrib;
    abbr        %Text.datatype;         #IMPLIED
    axis        CDATA                   #IMPLIED
    headers     IDREFS                  #IMPLIED
    %scope.attrib;
    rowspan     %Number.datatype;       '1'
    colspan     %Number.datatype;       '1'
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of td.attlist -->]]>

<!-- end of xhtml-basic-table-1.mod -->
```

Found in path(s):

* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-basic-table-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Common Attributes Module  ..................................... -->
<!-- file: xhtml-attribs-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-attribs-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Common Attributes 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-attribs-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Common Attributes

   This module declares many of the common attributes for the XHTML DTD.
   %NS.decl.attrib; is declared in the XHTML Qname module.

  Note that this file was extended in XHTML Modularization 1.1 to
  include declarations of "global" versions of the attribute collections.
  The global versions of the attributes are for use on elements in other
  namespaces.  The global version of "common" includes the xmlns declaration
  for the prefixed version of the xhtml namespace.  If you are only using a
  specific attribute or an individual attribute collection, you must also
  include the XHTML.xmlns.attrib.prefixed PE on your elements.
-->

<!ENTITY % id.attrib
   "id          ID                    #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.id.attrib
   "%XHTML.prefix;:id          ID       #IMPLIED"
>
]]>

<!ENTITY % class.attrib
   "class       NMTOKENS              #IMPLIED"
>
```

```
<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.class.attrib
   "%XHTML.prefix;:class       NMTOKENS              #IMPLIED"
>
]]>

<!ENTITY % title.attrib
   "title       %Text.datatype;         #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.title.attrib
   "%XHTML.prefix;:title       %Text.datatype;         #IMPLIED"
>
]]>

<!ENTITY % Core.extra.attrib "" >

<!ENTITY % Core.attrib
   "%XHTML.xmlns.attrib;
    %id.attrib;
    %class.attrib;
    %title.attrib;
    xml:space    ( preserve )           #FIXED 'preserve'
    %Core.extra.attrib;"
>

<!ENTITY % XHTML.global.core.extra.attrib "" >

<![%XHTML.global.attrs.prefixed;[

<!ENTITY % XHTML.global.core.attrib
   "%XHTML.global.id.attrib;
    %XHTML.global.class.attrib;
    %XHTML.global.title.attrib;
    %XHTML.global.core.extra.attrib;"
>
]]>

<!ENTITY % XHTML.global.core.attrib "" >


<!ENTITY % lang.attrib
   "xml:lang     %LanguageCode.datatype;  #IMPLIED"
>

<![%XHTML.bidi;[
```

```
<!ENTITY % dir.attrib
   "dir         ( ltr | rtl )          #IMPLIED"
>

<!ENTITY % I18n.attrib
   "%dir.attrib;
    %lang.attrib;"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY XHTML.global.i18n.attrib
   "%XHTML.prefix;:dir         ( ltr | rtl )          #IMPLIED
    %lang.attrib;"
>
]]>
<!ENTITY XHTML.global.i18n.attrib "" >

]]>
<!ENTITY % I18n.attrib
   "%lang.attrib;"
>
<!ENTITY % XHTML.global.i18n.attrib
   "%lang.attrib;"
>

<!ENTITY % Common.extra.attrib "" >
<!ENTITY % XHTML.global.common.extra.attrib "" >

<!-- intrinsic event attributes declared previously
-->
<!ENTITY % Events.attrib "" >

<!ENTITY % XHTML.global.events.attrib "" >

<!ENTITY % Common.attrib
   "%Core.attrib;
    %I18n.attrib;
    %Events.attrib;
    %Common.extra.attrib;"
>

<!ENTITY % XHTML.global.common.attrib
   "%XHTML.xmlns.attrib.prefixed;
    %XHTML.global.core.attrib;
  %XHTML.global.i18n.attrib;
  %XHTML.global.events.attrib;
  %XHTML.global.common.extra.attrib;"
>
```

<!-- end of xhtml-attribs-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-attribs-1.mod
No license file was found, but licenses were detected in source scan.

<!--
File isotech.ent produced by the XSL script entities.xsl
from input data in unicode.xml.

Copyright 1998 - 2009 W3C.

Use and distribution of this code are permitted under the terms of the
W3C Software Notice and License.
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

Please report any errors to David Carlisle
via the public W3C list www-math@w3.org.

The numeric character values assigned to each entity
(should) match the Unicode assignments in Unicode 5.x.

Entity names in this file are derived from files carrying the
following notice:

(C) International Organization for Standardization 1991
Permission to copy in any form is granted for use with
conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES General Technical//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isotech.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isotech PUBLIC
"-//W3C//ENTITIES General Technical//EN//XML"
"http://www.w3.org/2003/entities/2007/isotech.ent"

```
    >
    %isotech;

-->

<!ENTITY acd          "&#x0223F;" ><!--SINE WAVE -->
<!ENTITY aleph        "&#x02135;" ><!--ALEF SYMBOL -->
<!ENTITY and          "&#x02227;" ><!--LOGICAL AND -->
<!ENTITY And          "&#x02A53;" ><!--DOUBLE LOGICAL AND -->
<!ENTITY andand       "&#x02A55;" ><!--TWO INTERSECTING LOGICAL AND -->
<!ENTITY andd         "&#x02A5C;" ><!--LOGICAL AND WITH HORIZONTAL DASH -->
<!ENTITY andslope     "&#x02A58;" ><!--SLOPING LARGE AND -->
<!ENTITY andv         "&#x02A5A;" ><!--LOGICAL AND WITH MIDDLE STEM -->
<!ENTITY angrt        "&#x0221F;" ><!--RIGHT ANGLE -->
<!ENTITY angsph       "&#x02222;" ><!--SPHERICAL ANGLE -->
<!ENTITY angst        "&#x000C5;" ><!--LATIN CAPITAL LETTER A WITH RING ABOVE -->
<!ENTITY ap           "&#x02248;" ><!--ALMOST EQUAL TO -->
<!ENTITY apacir       "&#x02A6F;" ><!--ALMOST EQUAL TO WITH CIRCUMFLEX ACCENT -->
<!ENTITY awconint     "&#x02233;" ><!--ANTICLOCKWISE CONTOUR INTEGRAL -->
<!ENTITY awint        "&#x02A11;" ><!--ANTICLOCKWISE INTEGRATION -->
<!ENTITY becaus       "&#x02235;" ><!--BECAUSE -->
<!ENTITY bernou       "&#x0212C;" ><!--SCRIPT CAPITAL B -->
<!ENTITY bne          "&#x0003D;&#x020E5;" ><!--EQUALS SIGN with reverse slash -->
<!ENTITY bnequiv      "&#x02261;&#x020E5;" ><!--IDENTICAL TO with reverse slash -->
<!ENTITY bnot         "&#x02310;" ><!--REVERSED NOT SIGN -->
<!ENTITY bNot         "&#x02AED;" ><!--REVERSED DOUBLE STROKE NOT SIGN -->
<!ENTITY bottom       "&#x022A5;" ><!--UP TACK -->
<!ENTITY cap          "&#x02229;" ><!--INTERSECTION -->
<!ENTITY Cconint      "&#x02230;" ><!--VOLUME INTEGRAL -->
<!ENTITY cirfnint     "&#x02A10;" ><!--CIRCULATION FUNCTION -->
<!ENTITY compfn       "&#x02218;" ><!--RING OPERATOR -->
<!ENTITY cong         "&#x02245;" ><!--APPROXIMATELY EQUAL TO -->
<!ENTITY conint       "&#x0222E;" ><!--CONTOUR INTEGRAL -->
<!ENTITY Conint       "&#x0222F;" ><!--SURFACE INTEGRAL -->
<!ENTITY ctdot        "&#x022EF;" ><!--MIDLINE HORIZONTAL ELLIPSIS -->
<!ENTITY cup          "&#x0222A;" ><!--UNION -->
<!ENTITY cwconint     "&#x02232;" ><!--CLOCKWISE CONTOUR INTEGRAL -->
<!ENTITY cwint        "&#x02231;" ><!--CLOCKWISE INTEGRAL -->
<!ENTITY cylcty       "&#x0232D;" ><!--CYLINDRICITY -->
<!ENTITY disin        "&#x022F2;" ><!--ELEMENT OF WITH LONG HORIZONTAL STROKE -->
<!ENTITY Dot          "&#x000A8;" ><!--DIAERESIS -->
<!ENTITY DotDot       " &#x020DC;" ><!--COMBINING FOUR DOTS ABOVE -->
<!ENTITY dsol         "&#x029F6;" ><!--SOLIDUS WITH OVERBAR -->
<!ENTITY dtdot        "&#x022F1;" ><!--DOWN RIGHT DIAGONAL ELLIPSIS -->
<!ENTITY dwangle      "&#x029A6;" ><!--OBLIQUE ANGLE OPENING UP -->
<!ENTITY elinters     "&#x023E7;" ><!--ELECTRICAL INTERSECTION -->
<!ENTITY epar         "&#x022D5;" ><!--EQUAL AND PARALLEL TO -->
<!ENTITY eparsl       "&#x029E3;" ><!--EQUALS SIGN AND SLANTED PARALLEL -->
```

```
<!ENTITY equiv          "&#x02261;" ><!--IDENTICAL TO -->
<!ENTITY eqvparsl       "&#x029E5;" ><!--IDENTICAL TO AND SLANTED PARALLEL -->
<!ENTITY exist          "&#x02203;" ><!--THERE EXISTS -->
<!ENTITY fltns          "&#x025B1;" ><!--WHITE PARALLELOGRAM -->
<!ENTITY fnof           "&#x00192;" ><!--LATIN SMALL LETTER F WITH HOOK -->
<!ENTITY forall         "&#x02200;" ><!--FOR ALL -->
<!ENTITY fpartint       "&#x02A0D;" ><!--FINITE PART INTEGRAL -->
<!ENTITY ge             "&#x02265;" ><!--GREATER-THAN OR EQUAL TO -->
<!ENTITY hamilt         "&#x0210B;" ><!--SCRIPT CAPITAL H -->
<!ENTITY iff            "&#x021D4;" ><!--LEFT RIGHT DOUBLE ARROW -->
<!ENTITY iinfin         "&#x029DC;" ><!--INCOMPLETE INFINITY -->
<!ENTITY imped          "&#x001B5;" ><!--LATIN CAPITAL LETTER Z WITH STROKE -->
<!ENTITY infin          "&#x0221E;" ><!--INFINITY -->
<!ENTITY infintie       "&#x029DD;" ><!--TIE OVER INFINITY -->
<!ENTITY int            "&#x0222B;" ><!--INTEGRAL -->
<!ENTITY Int            "&#x0222C;" ><!--DOUBLE INTEGRAL -->
<!ENTITY intlarhk       "&#x02A17;" ><!--INTEGRAL WITH LEFTWARDS ARROW WITH HOOK -->
<!ENTITY isin           "&#x02208;" ><!--ELEMENT OF -->
<!ENTITY isindot        "&#x022F5;" ><!--ELEMENT OF WITH DOT ABOVE -->
<!ENTITY isinE          "&#x022F9;" ><!--ELEMENT OF WITH TWO HORIZONTAL STROKES -->
<!ENTITY isins          "&#x022F4;" ><!--SMALL ELEMENT OF WITH VERTICAL BAR AT END OF
HORIZONTAL STROKE -->
<!ENTITY isinsv         "&#x022F3;" ><!--ELEMENT OF WITH VERTICAL BAR AT END OF HORIZONTAL
STROKE -->
<!ENTITY isinv          "&#x02208;" ><!--ELEMENT OF -->
<!ENTITY lagran         "&#x02112;" ><!--SCRIPT CAPITAL L -->
<!ENTITY lang           "&#x027E8;" ><!--MATHEMATICAL LEFT ANGLE BRACKET -->
<!ENTITY Lang           "&#x027EA;" ><!--MATHEMATICAL LEFT DOUBLE ANGLE BRACKET -->
<!ENTITY lArr           "&#x021D0;" ><!--LEFTWARDS DOUBLE ARROW -->
<!ENTITY lbbrk          "&#x02772;" ><!--LIGHT LEFT TORTOISE SHELL BRACKET ORNAMENT -->
<!ENTITY le             "&#x02264;" ><!--LESS-THAN OR EQUAL TO -->
<!ENTITY loang          "&#x027EC;" ><!--MATHEMATICAL LEFT WHITE TORTOISE SHELL BRACKET --
>
<!ENTITY lobrk          "&#x027E6;" ><!--MATHEMATICAL LEFT WHITE SQUARE BRACKET -->
<!ENTITY lopar          "&#x02985;" ><!--LEFT WHITE PARENTHESIS -->
<!ENTITY lowast         "&#x02217;" ><!--ASTERISK OPERATOR -->
<!ENTITY minus          "&#x02212;" ><!--MINUS SIGN -->
<!ENTITY mnplus         "&#x02213;" ><!--MINUS-OR-PLUS SIGN -->
<!ENTITY nabla          "&#x02207;" ><!--NABLA -->
<!ENTITY ne             "&#x02260;" ><!--NOT EQUAL TO -->
<!ENTITY nedot          "&#x02250;&#x00338;" ><!--APPROACHES THE LIMIT with slash -->
<!ENTITY nhpar          "&#x02AF2;" ><!--PARALLEL WITH HORIZONTAL STROKE -->
<!ENTITY ni             "&#x0220B;" ><!--CONTAINS AS MEMBER -->
<!ENTITY nis            "&#x022FC;" ><!--SMALL CONTAINS WITH VERTICAL BAR AT END OF
HORIZONTAL STROKE -->
<!ENTITY nisd           "&#x022FA;" ><!--CONTAINS WITH LONG HORIZONTAL STROKE -->
<!ENTITY niv            "&#x0220B;" ><!--CONTAINS AS MEMBER -->
<!ENTITY Not            "&#x02AEC;" ><!--DOUBLE STROKE NOT SIGN -->
```

<!ENTITY notin "∉" ><!--NOT AN ELEMENT OF -->
<!ENTITY notindot "⋵̸" ><!--ELEMENT OF WITH DOT ABOVE with slash -->
<!ENTITY notinE "⋹̸" ><!--ELEMENT OF WITH TWO HORIZONTAL STROKES with slash -->
<!ENTITY notinva "∉" ><!--NOT AN ELEMENT OF -->
<!ENTITY notinvb "⋷" ><!--SMALL ELEMENT OF WITH OVERBAR -->
<!ENTITY notinvc "⋶" ><!--ELEMENT OF WITH OVERBAR -->
<!ENTITY notni "∌" ><!--DOES NOT CONTAIN AS MEMBER -->
<!ENTITY notniva "∌" ><!--DOES NOT CONTAIN AS MEMBER -->
<!ENTITY notnivb "⋾" ><!--SMALL CONTAINS WITH OVERBAR -->
<!ENTITY notnivc "⋽" ><!--CONTAINS WITH OVERBAR -->
<!ENTITY nparsl "⫽⃥" ><!--DOUBLE SOLIDUS OPERATOR with reverse slash -->
<!ENTITY npart "∂̸" ><!--PARTIAL DIFFERENTIAL with slash -->
<!ENTITY npolint "⨔" ><!--LINE INTEGRATION NOT INCLUDING THE POLE -->
<!ENTITY nvinfin "⧞" ><!--INFINITY NEGATED WITH VERTICAL BAR -->
<!ENTITY olcross "⦻" ><!--CIRCLE WITH SUPERIMPOSED X -->
<!ENTITY or "∨" ><!--LOGICAL OR -->
<!ENTITY Or "⩔" ><!--DOUBLE LOGICAL OR -->
<!ENTITY ord "⩝" ><!--LOGICAL OR WITH HORIZONTAL DASH -->
<!ENTITY order "ℴ" ><!--SCRIPT SMALL O -->
<!ENTITY oror "⩖" ><!--TWO INTERSECTING LOGICAL OR -->
<!ENTITY orslope "⩗" ><!--SLOPING LARGE OR -->
<!ENTITY orv "⩛" ><!--LOGICAL OR WITH MIDDLE STEM -->
<!ENTITY par "∥" ><!--PARALLEL TO -->
<!ENTITY parsl "⫽" ><!--DOUBLE SOLIDUS OPERATOR -->
<!ENTITY part "∂" ><!--PARTIAL DIFFERENTIAL -->
<!ENTITY permil "‰" ><!--PER MILLE SIGN -->
<!ENTITY perp "⊥" ><!--UP TACK -->
<!ENTITY pertenk "‱" ><!--PER TEN THOUSAND SIGN -->
<!ENTITY phmmat "ℳ" ><!--SCRIPT CAPITAL M -->
<!ENTITY pointint "⨕" ><!--INTEGRAL AROUND A POINT OPERATOR -->
<!ENTITY prime "′" ><!--PRIME -->
<!ENTITY Prime "″" ><!--DOUBLE PRIME -->
<!ENTITY profalar "⌮" ><!--ALL AROUND-PROFILE -->
<!ENTITY profline "⌒" ><!--ARC -->
<!ENTITY profsurf "⌓" ><!--SEGMENT -->
<!ENTITY prop "∝" ><!--PROPORTIONAL TO -->
<!ENTITY qint "⨌" ><!--QUADRUPLE INTEGRAL OPERATOR -->
<!ENTITY qprime "⁗" ><!--QUADRUPLE PRIME -->
<!ENTITY quatint "⨖" ><!--QUATERNION INTEGRAL OPERATOR -->
<!ENTITY radic "√" ><!--SQUARE ROOT -->
<!ENTITY rang "⟩" ><!--MATHEMATICAL RIGHT ANGLE BRACKET -->
<!ENTITY Rang "⟫" ><!--MATHEMATICAL RIGHT DOUBLE ANGLE BRACKET -->
<!ENTITY rArr "⇒" ><!--RIGHTWARDS DOUBLE ARROW -->
<!ENTITY rbbrk "❳" ><!--LIGHT RIGHT TORTOISE SHELL BRACKET ORNAMENT -->
<!ENTITY roang "⟭" ><!--MATHEMATICAL RIGHT WHITE TORTOISE SHELL BRACKET -->
<!ENTITY robrk "⟧" ><!--MATHEMATICAL RIGHT WHITE SQUARE BRACKET -->

<!ENTITY ropar "⦆" ><!--RIGHT WHITE PARENTHESIS -->
<!ENTITY rppolint "⨒" ><!--LINE INTEGRATION WITH RECTANGULAR PATH AROUND POLE -->
<!ENTITY scpolint "⨓" ><!--LINE INTEGRATION WITH SEMICIRCULAR PATH AROUND POLE -->
<!ENTITY sim "∼" ><!--TILDE OPERATOR -->
<!ENTITY simdot "⩪" ><!--TILDE OPERATOR WITH DOT ABOVE -->
<!ENTITY sime "≃" ><!--ASYMPTOTICALLY EQUAL TO -->
<!ENTITY smeparsl "⧤" ><!--EQUALS SIGN AND SLANTED PARALLEL WITH TILDE ABOVE -->
<!ENTITY square "□" ><!--WHITE SQUARE -->
<!ENTITY squarf "▪" ><!--BLACK SMALL SQUARE -->
<!ENTITY strns "¯" ><!--MACRON -->
<!ENTITY sub "⊂" ><!--SUBSET OF -->
<!ENTITY sube "⊆" ><!--SUBSET OF OR EQUAL TO -->
<!ENTITY sup "⊃" ><!--SUPERSET OF -->
<!ENTITY supe "⊇" ><!--SUPERSET OF OR EQUAL TO -->
<!ENTITY tdot " ⃛" ><!--COMBINING THREE DOTS ABOVE -->
<!ENTITY there4 "∴" ><!--THEREFORE -->
<!ENTITY tint "∭" ><!--TRIPLE INTEGRAL -->
<!ENTITY top "⊤" ><!--DOWN TACK -->
<!ENTITY topbot "⌶" ><!--APL FUNCTIONAL SYMBOL I-BEAM -->
<!ENTITY topcir "⫱" ><!--DOWN TACK WITH CIRCLE BELOW -->
<!ENTITY tprime "‴" ><!--TRIPLE PRIME -->
<!ENTITY utdot "⋰" ><!--UP RIGHT DIAGONAL ELLIPSIS -->
<!ENTITY uwangle "⦧" ><!--OBLIQUE ANGLE OPENING DOWN -->
<!ENTITY vangrt "⦜" ><!--RIGHT ANGLE VARIANT WITH SQUARE -->
<!ENTITY veeeq "≚" ><!--EQUIANGULAR TO -->
<!ENTITY Verbar "‖" ><!--DOUBLE VERTICAL LINE -->
<!ENTITY wedgeq "≙" ><!--ESTIMATES -->
<!ENTITY xnis "⋻" ><!--CONTAINS WITH VERTICAL BAR AT END OF HORIZONTAL STROKE -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isotech.ent
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Cursor Module ..... -->
<!-- file: svg-cursor.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-cursor.mod,v 1.3 2002/10/24 17:40:15 fujisawa Exp $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

```
      PUBLIC "-//W3C//ELEMENTS SVG 1.1 Cursor//EN"
      SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-cursor.mod"

    ..................................................................... -->

<!-- Cursor

      cursor

    This module declares markup to provide support for cursor.
-->

<!-- 'cursor' property/attribute value (e.g., 'crosshair', <uri>) -->
<!ENTITY % CursorValue.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.cursor.qname "cursor" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.XLinkRequired.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Cursor.class ..... -->

<!ENTITY % SVG.Cursor.extra.class "" >

<!ENTITY % SVG.Cursor.class
   "| %SVG.cursor.qname; %SVG.Cursor.extra.class;"
>

<!-- SVG.Cursor.attrib ..... -->

<!ENTITY % SVG.Cursor.extra.attrib "" >

<!ENTITY % SVG.Cursor.attrib
   "cursor %CursorValue.datatype; #IMPLIED
    %SVG.Cursor.extra.attrib;"
>

<!-- cursor: Cursor Element ..... -->

<!ENTITY % SVG.cursor.extra.content "" >

<!ENTITY % SVG.cursor.element "INCLUDE" >
```

```
<![%SVG.cursor.element;[
<!ENTITY % SVG.cursor.content
  "( %SVG.Description.class; %SVG.cursor.extra.content; )*"
>
<!ELEMENT %SVG.cursor.qname; %SVG.cursor.content; >
<!-- end of SVG.cursor.element -->]]>

<!ENTITY % SVG.cursor.attlist "INCLUDE" >
<![%SVG.cursor.attlist;[
<!ATTLIST %SVG.cursor.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.XLinkRequired.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
>
<!-- end of SVG.cursor.attlist -->]]>

<!-- end of svg-cursor.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-cursor.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Table Module  ..................................................... -->
<!-- file: xhtml-table-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-table-1.mod,v 4.1 2001/04/10 09:42:30 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Tables 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-table-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Tables

      table, caption, thead, tfoot, tbody, colgroup, col, tr, th, td

   This module declares element types and attributes used to provide
```

```
   table markup similar to HTML 4, including features that enable
   better accessibility for non-visual user agents.
-->

<!-- declare qualified element type names:
-->
<!ENTITY % table.qname  "table" >
<!ENTITY % caption.qname  "caption" >
<!ENTITY % thead.qname  "thead" >
<!ENTITY % tfoot.qname  "tfoot" >
<!ENTITY % tbody.qname  "tbody" >
<!ENTITY % colgroup.qname  "colgroup" >
<!ENTITY % col.qname  "col" >
<!ENTITY % tr.qname  "tr" >
<!ENTITY % th.qname  "th" >
<!ENTITY % td.qname  "td" >

<!-- The frame attribute specifies which parts of the frame around
   the table should be rendered. The values are not the same as
   CALS to avoid a name clash with the valign attribute.
-->
<!ENTITY % frame.attrib
   "frame       ( void
              | above
              | below
              | hsides
              | lhs
              | rhs
              | vsides
              | box
              | border )             #IMPLIED"
>

<!-- The rules attribute defines which rules to draw between cells:

   If rules is absent then assume:

     "none" if border is absent or border="0" otherwise "all"
-->
<!ENTITY % rules.attrib
   "rules       ( none
              | groups
              | rows
              | cols
              | all )               #IMPLIED"
>

<!-- horizontal alignment attributes for cell contents
```

```
-->
<!ENTITY % CellHAlign.attrib
   "align       ( left
                | center
                | right
                | justify
                | char )              #IMPLIED
    char        %Character.datatype;    #IMPLIED
    charoff     %Length.datatype;       #IMPLIED"
>

<!-- vertical alignment attribute for cell contents
-->
<!ENTITY % CellVAlign.attrib
   "valign      ( top
                | middle
                | bottom
                | baseline )           #IMPLIED"
>

<!-- scope is simpler than axes attribute for common tables
-->
<!ENTITY % scope.attrib
   "scope       ( row
                | col
                | rowgroup
                | colgroup )           #IMPLIED"
>

<!-- table: Table Element ..... -->

<!ENTITY % table.element  "INCLUDE" >
<![%table.element;[
<!ENTITY % table.content
   "( %caption.qname;?, ( %col.qname;* | %colgroup.qname;* ),
    (( %thead.qname;?, %tfoot.qname;?, %tbody.qname;+ ) | ( %tr.qname;+ )))"
>
<!ELEMENT %table.qname;  %table.content; >
<!-- end of table.element -->]]>

<!ENTITY % table.attlist  "INCLUDE" >
<![%table.attlist;[
<!ATTLIST %table.qname;
    %Common.attrib;
    summary     %Text.datatype;         #IMPLIED
    width       %Length.datatype;       #IMPLIED
    border      %Pixels.datatype;       #IMPLIED
    %frame.attrib;
```

```
    %rules.attrib;
    cellspacing  %Length.datatype;        #IMPLIED
    cellpadding  %Length.datatype;        #IMPLIED
>
<!-- end of table.attlist -->]]>

<!-- caption: Table Caption ..... -->

<!ENTITY % caption.element  "INCLUDE" >
<![%caption.element;[
<!ENTITY % caption.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ELEMENT %caption.qname;  %caption.content; >
<!-- end of caption.element -->]]>

<!ENTITY % caption.attlist  "INCLUDE" >
<![%caption.attlist;[
<!ATTLIST %caption.qname;
    %Common.attrib;
>
<!-- end of caption.attlist -->]]>

<!-- thead: Table Header ..... -->

<!-- Use thead to duplicate headers when breaking table
   across page boundaries, or for static headers when
   tbody sections are rendered in scrolling panel.
-->

<!ENTITY % thead.element  "INCLUDE" >
<![%thead.element;[
<!ENTITY % thead.content  "( %tr.qname; )+" >
<!ELEMENT %thead.qname;  %thead.content; >
<!-- end of thead.element -->]]>

<!ENTITY % thead.attlist  "INCLUDE" >
<![%thead.attlist;[
<!ATTLIST %thead.qname;
    %Common.attrib;
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of thead.attlist -->]]>

<!-- tfoot: Table Footer ..... -->

<!-- Use tfoot to duplicate footers when breaking table
```

```
   across page boundaries, or for static footers when
   tbody sections are rendered in scrolling panel.
-->

<!ENTITY % tfoot.element  "INCLUDE" >
<![%tfoot.element;[
<!ENTITY % tfoot.content  "( %tr.qname; )+" >
<!ELEMENT %tfoot.qname;  %tfoot.content; >
<!-- end of tfoot.element -->]]>

<!ENTITY % tfoot.attlist  "INCLUDE" >
<![%tfoot.attlist;[
<!ATTLIST %tfoot.qname;
    %Common.attrib;
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of tfoot.attlist -->]]>

<!-- tbody: Table Body ..... -->

<!-- Use multiple tbody sections when rules are needed
   between groups of table rows.
-->

<!ENTITY % tbody.element  "INCLUDE" >
<![%tbody.element;[
<!ENTITY % tbody.content  "( %tr.qname; )+" >
<!ELEMENT %tbody.qname;  %tbody.content; >
<!-- end of tbody.element -->]]>

<!ENTITY % tbody.attlist  "INCLUDE" >
<![%tbody.attlist;[
<!ATTLIST %tbody.qname;
    %Common.attrib;
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of tbody.attlist -->]]>

<!-- colgroup: Table Column Group ..... -->

<!-- colgroup groups a set of col elements. It allows you
   to group several semantically-related columns together.
-->

<!ENTITY % colgroup.element  "INCLUDE" >
<![%colgroup.element;[
```

```
<!ENTITY % colgroup.content  "( %col.qname; )*" >
<!ELEMENT %colgroup.qname;  %colgroup.content; >
<!-- end of colgroup.element -->]]>

<!ENTITY % colgroup.attlist  "INCLUDE" >
<![%colgroup.attlist;[
<!ATTLIST %colgroup.qname;
    %Common.attrib;
    span        %Number.datatype;       '1'
    width       %MultiLength.datatype;   #IMPLIED
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of colgroup.attlist -->]]>

<!-- col: Table Column ..... -->

<!-- col elements define the alignment properties for
   cells in one or more columns.

   The width attribute specifies the width of the
   columns, e.g.

     width="64"        width in screen pixels
     width="0.5*"      relative width of 0.5

   The span attribute causes the attributes of one
   col element to apply to more than one column.
-->

<!ENTITY % col.element  "INCLUDE" >
<![%col.element;[
<!ENTITY % col.content  "EMPTY" >
<!ELEMENT %col.qname;  %col.content; >
<!-- end of col.element -->]]>

<!ENTITY % col.attlist  "INCLUDE" >
<![%col.attlist;[
<!ATTLIST %col.qname;
    %Common.attrib;
    span        %Number.datatype;       '1'
    width       %MultiLength.datatype;   #IMPLIED
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of col.attlist -->]]>

<!-- tr: Table Row ..... -->
```

```
<!ENTITY % tr.element  "INCLUDE" >
<![%tr.element;[
<!ENTITY % tr.content  "( %th.qname; | %td.qname; )+" >
<!ELEMENT %tr.qname;  %tr.content; >
<!-- end of tr.element -->]]>

<!ENTITY % tr.attlist  "INCLUDE" >
<![%tr.attlist;[
<!ATTLIST %tr.qname;
    %Common.attrib;
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of tr.attlist -->]]>

<!-- th: Table Header Cell ..... -->

<!-- th is for header cells, td for data,
   but for cells acting as both use td
-->

<!ENTITY % th.element  "INCLUDE" >
<![%th.element;[
<!ENTITY % th.content
   "( #PCDATA | %Flow.mix; )*"
>
<!ELEMENT %th.qname;  %th.content; >
<!-- end of th.element -->]]>

<!ENTITY % th.attlist  "INCLUDE" >
<![%th.attlist;[
<!ATTLIST %th.qname;
    %Common.attrib;
    abbr        %Text.datatype;         #IMPLIED
    axis        CDATA                   #IMPLIED
    headers     IDREFS                  #IMPLIED
    %scope.attrib;
    rowspan     %Number.datatype;       '1'
    colspan     %Number.datatype;       '1'
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of th.attlist -->]]>

<!-- td: Table Data Cell ..... -->

<!ENTITY % td.element  "INCLUDE" >
```

```
<![%td.element;[
<!ENTITY % td.content
   "( #PCDATA | %Flow.mix; )*"
>
<!ELEMENT %td.qname;  %td.content; >
<!-- end of td.element -->]]>

<!ENTITY % td.attlist  "INCLUDE" >
<![%td.attlist;[
<!ATTLIST %td.qname;
    %Common.attrib;
    abbr        %Text.datatype;         #IMPLIED
    axis        CDATA                   #IMPLIED
    headers     IDREFS                  #IMPLIED
    %scope.attrib;
    rowspan     %Number.datatype;       '1'
    colspan     %Number.datatype;       '1'
    %CellHAlign.attrib;
    %CellVAlign.attrib;
>
<!-- end of td.attlist -->]]>

<!-- end of xhtml-table-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-table-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Simplified Forms Module  ....................................... -->
<!-- file: xhtml-basic-form-1.mod

   This is XHTML Basic, a proper subset of XHTML.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-basic-form-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Basic Forms 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-basic-form-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Basic Forms
```

```
   This forms module is based on the HTML 3.2 forms model, with
   the WAI-requested addition of the label element. While this
   module essentially mimics the content model and attributes of
   HTML 3.2 forms, the element types declared herein also include
   all HTML 4 common attributes.

      form, label, input, select, option, textarea
-->

<!-- declare qualified element type names:
-->
<!ENTITY % form.qname  "form" >
<!ENTITY % label.qname  "label" >
<!ENTITY % input.qname  "input" >
<!ENTITY % select.qname  "select" >
<!ENTITY % option.qname  "option" >
<!ENTITY % textarea.qname  "textarea" >

<!-- %BlkNoForm.mix; includes all non-form block elements,
   plus %Misc.class;
-->
<!ENTITY % BlkNoForm.mix
   "%Heading.class;
    | %List.class;
    | %BlkStruct.class;
    %BlkPhras.class;
    %BlkPres.class;
    | %table.qname;
    %Block.extra;
    %Misc.class;"
>

<!-- form: Form Element ..................................... -->

<!ENTITY % form.element  "INCLUDE" >
<![%form.element;[
<!ENTITY % form.content
   "( %BlkNoForm.mix; )+"
>
<!ELEMENT %form.qname;  %form.content; >
<!-- end of form.element -->]]>

<!ENTITY % form.attlist  "INCLUDE" >
<![%form.attlist;[
<!ATTLIST %form.qname;
    %Common.attrib;
    action      %URI.datatype;          #REQUIRED
    method      ( get | post )          'get'
```

```
    enctype      %ContentType.datatype;   'application/x-www-form-urlencoded'
>
<!-- end of form.attlist -->]]>

<!-- label: Form Field Label Text ..... -->

<!ENTITY % label.element  "INCLUDE" >
<![%label.element;[
<!-- Each label must not contain more than ONE field
-->
<!ENTITY % label.content
   "( #PCDATA
    | %input.qname; | %select.qname; | %textarea.qname;
    | %InlStruct.class;
    %InlPhras.class;
    %I18n.class;
    %InlPres.class;
    %InlSpecial.class;
    %Misc.class; )*"
>
<!ELEMENT %label.qname;  %label.content; >
<!-- end of label.element -->]]>

<!ENTITY % label.attlist  "INCLUDE" >
<![%label.attlist;[
<!ATTLIST %label.qname;
    %Common.attrib;
    for          IDREF                #IMPLIED
    accesskey    %Character.datatype;     #IMPLIED
>
<!-- end of label.attlist -->]]>

<!-- input: Form Control ..... -->

<!ENTITY % input.element  "INCLUDE" >
<![%input.element;[
<!ENTITY % input.content  "EMPTY" >
<!ELEMENT %input.qname;  %input.content; >
<!-- end of input.element -->]]>

<!-- Basic Forms removes 'image' and 'file' input types.
-->
<!ENTITY % input.attlist  "INCLUDE" >
<![%input.attlist;[
<!ENTITY % InputType.class
   "( text | password | checkbox | radio
    | submit | reset | hidden )"
>
```

```
<!-- attribute name required for all but submit & reset
-->
<!ATTLIST %input.qname;
    %Common.attrib;
    type        %InputType.class;       'text'
    name        CDATA                   #IMPLIED
    value       CDATA                   #IMPLIED
    checked     ( checked )             #IMPLIED
    size        CDATA                   #IMPLIED
    maxlength   %Number.datatype;       #IMPLIED
    src         %URI.datatype;          #IMPLIED
    accesskey   %Character.datatype;    #IMPLIED
>
<!-- end of input.attlist -->]]>

<!-- select: Option Selector ..... -->

<!ENTITY % select.element  "INCLUDE" >
<![%select.element;[
<!ENTITY % select.content  "( %option.qname; )+" >
<!ELEMENT %select.qname;  %select.content; >
<!-- end of select.element -->]]>

<!ENTITY % select.attlist  "INCLUDE" >
<![%select.attlist;[
<!ATTLIST %select.qname;
    %Common.attrib;
    name        CDATA                   #IMPLIED
    size        %Number.datatype;       #IMPLIED
    multiple    ( multiple )            #IMPLIED
>
<!-- end of select.attlist -->]]>

<!-- option: Selectable Choice ..... -->

<!ENTITY % option.element  "INCLUDE" >
<![%option.element;[
<!ENTITY % option.content  "( #PCDATA )" >
<!ELEMENT %option.qname;  %option.content; >
<!-- end of option.element -->]]>

<!ENTITY % option.attlist  "INCLUDE" >
<![%option.attlist;[
<!ATTLIST %option.qname;
    %Common.attrib;
    selected    ( selected )            #IMPLIED
    value       CDATA                   #IMPLIED
>
```

```
<!-- end of option.attlist -->]]>

<!-- textarea: Multi-Line Text Field ..... -->

<!ENTITY % textarea.element  "INCLUDE" >
<![%textarea.element;[
<!ENTITY % textarea.content  "( #PCDATA )" >
<!ELEMENT %textarea.qname;  %textarea.content; >
<!-- end of textarea.element -->]]>

<!ENTITY % textarea.attlist  "INCLUDE" >
<![%textarea.attlist;[
<!ATTLIST %textarea.qname;
    %Common.attrib;
    name         CDATA                  #IMPLIED
    rows         %Number.datatype;      #REQUIRED
    cols         %Number.datatype;      #REQUIRED
    accesskey    %Character.datatype;   #IMPLIED
>
<!-- end of textarea.attlist -->]]>

<!-- end of xhtml-basic-form-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-basic-form-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 XLink Attribute Module ..................................... -->
<!-- file: svg-xlink-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-xlink-attrib.mod,v 1.2 2002/04/20 18:07:43 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 XLink Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-xlink-attrib.mod"

   ..................................................................... -->

<!-- XLink Attribute

     type, href, role, arcrole, title, show, actuate

   This module defines the XLink, XLinkRequired, XLinkEmbed, and
```

```
    XLinkReplace attribute set.
-->

<!ENTITY % SVG.XLink.extra.attrib "" >

<!ENTITY % SVG.XLink.attrib
  "%XLINK.xmlns.attrib;
   %XLINK.pfx;type ( simple ) #FIXED 'simple'
   %XLINK.pfx;href %URI.datatype; #IMPLIED
   %XLINK.pfx;role %URI.datatype; #IMPLIED
   %XLINK.pfx;arcrole %URI.datatype; #IMPLIED
   %XLINK.pfx;title CDATA #IMPLIED
   %XLINK.pfx;show ( other ) 'other'
   %XLINK.pfx;actuate ( onLoad ) #FIXED 'onLoad'
   %SVG.XLink.extra.attrib;"
>

<!ENTITY % SVG.XLinkRequired.extra.attrib "" >

<!ENTITY % SVG.XLinkRequired.attrib
  "%XLINK.xmlns.attrib;
   %XLINK.pfx;type ( simple ) #FIXED 'simple'
   %XLINK.pfx;href %URI.datatype; #REQUIRED
   %XLINK.pfx;role %URI.datatype; #IMPLIED
   %XLINK.pfx;arcrole %URI.datatype; #IMPLIED
   %XLINK.pfx;title CDATA #IMPLIED
   %XLINK.pfx;show ( other ) 'other'
   %XLINK.pfx;actuate ( onLoad ) #FIXED 'onLoad'
   %SVG.XLinkRequired.extra.attrib;"
>

<!ENTITY % SVG.XLinkEmbed.extra.attrib "" >

<!ENTITY % SVG.XLinkEmbed.attrib
  "%XLINK.xmlns.attrib;
   %XLINK.pfx;type ( simple ) #FIXED 'simple'
   %XLINK.pfx;href %URI.datatype; #REQUIRED
   %XLINK.pfx;role %URI.datatype; #IMPLIED
   %XLINK.pfx;arcrole %URI.datatype; #IMPLIED
   %XLINK.pfx;title CDATA #IMPLIED
   %XLINK.pfx;show ( embed ) 'embed'
   %XLINK.pfx;actuate ( onLoad ) #FIXED 'onLoad'
   %SVG.XLinkEmbed.extra.attrib;"
>

<!ENTITY % SVG.XLinkReplace.extra.attrib "" >

<!ENTITY % SVG.XLinkReplace.attrib
```

```
  "%XLINK.xmlns.attrib;
   %XLINK.pfx;type ( simple ) #FIXED 'simple'
   %XLINK.pfx;href %URI.datatype; #REQUIRED
   %XLINK.pfx;role %URI.datatype; #IMPLIED
   %XLINK.pfx;arcrole %URI.datatype; #IMPLIED
   %XLINK.pfx;title CDATA #IMPLIED
   %XLINK.pfx;show ( new | replace ) 'replace'
   %XLINK.pfx;actuate ( onRequest ) #FIXED 'onRequest'
   %SVG.XLinkReplace.extra.attrib;"
>

<!-- end of svg-xlink-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-xlink-attrib.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- MathML Qualified Names Module  ..................................... -->
<!-- file: mathml2-qname-1.mod

   This is the Mathematical Markup Language (MathML) 2.0, an XML
   application for describing mathematical notation and capturing
   both its structure and content.

   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: mathml2-qname-1.mod,v 1.2 2003/04/08 00:11:16 davidc Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES MathML 2.0 Qualified Names 1.0//EN"
     SYSTEM "mathml2-qname-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- MathML Qualified Names

   This module is contained in two parts, labeled Section 'A' and 'B':

     Section A declares parameter entities to support namespace-
     qualified names, namespace declarations, and name prefixing
     for MathML.

     Section B declares parameter entities used to provide
     namespace-qualified names for all MathML element types.
```

```
   This module is derived from the XHTML Qualified Names Template module.
-->

<!-- Section A: XHTML XML Namespace Framework :::::::::::::::::::: -->

<!ENTITY % NS.prefixed     "IGNORE" >
<!ENTITY % MATHML.prefixed "%NS.prefixed;" >

<!-- XLink ..... -->

<!ENTITY % XLINK.prefix        "xlink" >
<!ENTITY % XLINK.xmlns "http://www.w3.org/1999/xlink" >
<!ENTITY % XLINK.xmlns.attrib
   "xmlns:%XLINK.prefix;  CDATA          #FIXED '%XLINK.xmlns;'"
>

<!-- W3C XML Schema ..... -->

<!ENTITY % Schema.prefix        "xsi" >
<!ENTITY % Schema.xmlns "http://www.w3.org/2001/XMLSchema-instance" >
<!ENTITY % Schema.xmlns.attrib
   "xmlns:%Schema.prefix;  CDATA          #IMPLIED"
>

<!-- MathML ..... -->

<!ENTITY % MATHML.xmlns    "http://www.w3.org/1998/Math/MathML" >
<!ENTITY % MATHML.prefix   "m" >
<![%MATHML.prefixed;[
<!ENTITY % MATHML.xmlns.extra.attrib  "" >
]]>
<!ENTITY % MATHML.xmlns.extra.attrib
   "%XLINK.xmlns.attrib;
    %Schema.xmlns.attrib;" >

<![%MATHML.prefixed;[
<!ENTITY % MATHML.pfx  "%MATHML.prefix;:" >
<!ENTITY % MATHML.xmlns.attrib
   "xmlns:%MATHML.prefix;  CDATA   #FIXED '%MATHML.xmlns;'
    %MATHML.xmlns.extra.attrib;"
>
]]>
<!ENTITY % MATHML.pfx  "" >
<!ENTITY % MATHML.xmlns.attrib
   "xmlns       CDATA          #FIXED '%MATHML.xmlns;'
    %MATHML.xmlns.extra.attrib;"
>
```

```
<![%NS.prefixed;[
<!ENTITY % XHTML.xmlns.extra.attrib
   "%MATHML.xmlns.attrib;" >
]]>
<!ENTITY % XHTML.xmlns.extra.attrib
   "%XLINK.xmlns.attrib;
    %Schema.xmlns.attrib;"
>

<!-- Section B: MathML Qualified Names ::::::::::::::::::::::::::::: -->

<!-- 9. This section declares parameter entities used to provide
      namespace-qualified names for all MathML element types.
-->

<!ENTITY % abs.qname            "%MATHML.pfx;abs" >
<!ENTITY % and.qname            "%MATHML.pfx;and" >
<!ENTITY % annotation-xml.qname "%MATHML.pfx;annotation-xml" >
<!ENTITY % annotation.qname     "%MATHML.pfx;annotation" >
<!ENTITY % apply.qname          "%MATHML.pfx;apply" >
<!ENTITY % approx.qname         "%MATHML.pfx;approx" >
<!ENTITY % arccos.qname         "%MATHML.pfx;arccos" >
<!ENTITY % arccosh.qname        "%MATHML.pfx;arccosh" >
<!ENTITY % arccosh.qname        "%MATHML.pfx;arccosh" >
<!ENTITY % arccot.qname         "%MATHML.pfx;arccot" >
<!ENTITY % arccoth.qname        "%MATHML.pfx;arccoth" >
<!ENTITY % arccsc.qname         "%MATHML.pfx;arccsc" >
<!ENTITY % arccsch.qname        "%MATHML.pfx;arccsch" >
<!ENTITY % arcsec.qname         "%MATHML.pfx;arcsec" >
<!ENTITY % arcsech.qname        "%MATHML.pfx;arcsech" >
<!ENTITY % arcsin.qname         "%MATHML.pfx;arcsin" >
<!ENTITY % arcsinh.qname        "%MATHML.pfx;arcsinh" >
<!ENTITY % arctan.qname         "%MATHML.pfx;arctan" >
<!ENTITY % arctanh.qname        "%MATHML.pfx;arctanh" >
<!ENTITY % arg.qname            "%MATHML.pfx;arg" >
<!ENTITY % bvar.qname           "%MATHML.pfx;bvar" >
<!ENTITY % card.qname           "%MATHML.pfx;card" >
<!ENTITY % cartesianproduct.qname "%MATHML.pfx;cartesianproduct" >
<!ENTITY % ceiling.qname        "%MATHML.pfx;ceiling" >
<!ENTITY % ci.qname             "%MATHML.pfx;ci" >
<!ENTITY % cn.qname             "%MATHML.pfx;cn" >
<!ENTITY % codomain.qname       "%MATHML.pfx;codomain" >
<!ENTITY % complexes.qname      "%MATHML.pfx;complexes" >
<!ENTITY % compose.qname        "%MATHML.pfx;compose" >
<!ENTITY % condition.qname      "%MATHML.pfx;condition" >
<!ENTITY % conjugate.qname      "%MATHML.pfx;conjugate" >
<!ENTITY % cos.qname            "%MATHML.pfx;cos" >
```

```
<!ENTITY % cosh.qname          "%MATHML.pfx;cosh" >
<!ENTITY % cot.qname           "%MATHML.pfx;cot" >
<!ENTITY % coth.qname          "%MATHML.pfx;coth" >
<!ENTITY % csc.qname           "%MATHML.pfx;csc" >
<!ENTITY % csch.qname          "%MATHML.pfx;csch" >
<!ENTITY % csymbol.qname       "%MATHML.pfx;csymbol" >
<!ENTITY % curl.qname          "%MATHML.pfx;curl" >
<!ENTITY % declare.qname       "%MATHML.pfx;declare" >
<!ENTITY % degree.qname        "%MATHML.pfx;degree" >
<!ENTITY % determinant.qname   "%MATHML.pfx;determinant" >
<!ENTITY % diff.qname          "%MATHML.pfx;diff" >
<!ENTITY % divergence.qname    "%MATHML.pfx;divergence" >
<!ENTITY % divide.qname        "%MATHML.pfx;divide" >
<!ENTITY % domain.qname        "%MATHML.pfx;domain" >
<!ENTITY % domainofapplication.qname "%MATHML.pfx;domainofapplication" >
<!ENTITY % emptyset.qname      "%MATHML.pfx;emptyset" >
<!ENTITY % eq.qname            "%MATHML.pfx;eq" >
<!ENTITY % equivalent.qname    "%MATHML.pfx;equivalent" >
<!ENTITY % eulergamma.qname    "%MATHML.pfx;eulergamma" >
<!ENTITY % exists.qname        "%MATHML.pfx;exists" >
<!ENTITY % exp.qname           "%MATHML.pfx;exp" >
<!ENTITY % exponentiale.qname  "%MATHML.pfx;exponentiale" >
<!ENTITY % factorial.qname     "%MATHML.pfx;factorial" >
<!ENTITY % factorof.qname      "%MATHML.pfx;factorof" >
<!ENTITY % false.qname         "%MATHML.pfx;false" >
<!ENTITY % floor.qname         "%MATHML.pfx;floor" >
<!ENTITY % fn.qname            "%MATHML.pfx;fn" >
<!ENTITY % forall.qname        "%MATHML.pfx;forall" >
<!ENTITY % gcd.qname           "%MATHML.pfx;gcd" >
<!ENTITY % geq.qname           "%MATHML.pfx;geq" >
<!ENTITY % grad.qname          "%MATHML.pfx;grad" >
<!ENTITY % gt.qname            "%MATHML.pfx;gt" >
<!ENTITY % ident.qname         "%MATHML.pfx;ident" >
<!ENTITY % image.qname         "%MATHML.pfx;image" >
<!ENTITY % imaginary.qname     "%MATHML.pfx;imaginary" >
<!ENTITY % imaginaryi.qname    "%MATHML.pfx;imaginaryi" >
<!ENTITY % implies.qname       "%MATHML.pfx;implies" >
<!ENTITY % in.qname            "%MATHML.pfx;in" >
<!ENTITY % infinity.qname      "%MATHML.pfx;infinity" >
<!ENTITY % int.qname           "%MATHML.pfx;int" >
<!ENTITY % integers.qname      "%MATHML.pfx;integers" >
<!ENTITY % intersect.qname     "%MATHML.pfx;intersect" >
<!ENTITY % interval.qname      "%MATHML.pfx;interval" >
<!ENTITY % inverse.qname       "%MATHML.pfx;inverse" >
<!ENTITY % lambda.qname        "%MATHML.pfx;lambda" >
<!ENTITY % laplacian.qname     "%MATHML.pfx;laplacian" >
<!ENTITY % lcm.qname           "%MATHML.pfx;lcm" >
<!ENTITY % leq.qname           "%MATHML.pfx;leq" >
```

```
<!ENTITY % limit.qname          "%MATHML.pfx;limit" >
<!ENTITY % list.qname           "%MATHML.pfx;list" >
<!ENTITY % ln.qname             "%MATHML.pfx;ln" >
<!ENTITY % log.qname            "%MATHML.pfx;log" >
<!ENTITY % logbase.qname        "%MATHML.pfx;logbase" >
<!ENTITY % lowlimit.qname       "%MATHML.pfx;lowlimit" >
<!ENTITY % lt.qname             "%MATHML.pfx;lt" >
<!ENTITY % maction.qname        "%MATHML.pfx;maction" >
<!ENTITY % malignngroup.qname   "%MATHML.pfx;malignngroup" >
<!ENTITY % malignmark.qname     "%MATHML.pfx;malignmark" >
<!ENTITY % math.qname           "%MATHML.pfx;math" >
<!ENTITY % matrix.qname         "%MATHML.pfx;matrix" >
<!ENTITY % matrixrow.qname      "%MATHML.pfx;matrixrow" >
<!ENTITY % max.qname            "%MATHML.pfx;max" >
<!ENTITY % mean.qname           "%MATHML.pfx;mean" >
<!ENTITY % median.qname         "%MATHML.pfx;median" >
<!ENTITY % menclose.qname       "%MATHML.pfx;menclose" >
<!ENTITY % merror.qname         "%MATHML.pfx;merror" >
<!ENTITY % mfenced.qname        "%MATHML.pfx;mfenced" >
<!ENTITY % mfrac.qname          "%MATHML.pfx;mfrac" >
<!ENTITY % mglyph.qname         "%MATHML.pfx;mglyph" >
<!ENTITY % mi.qname             "%MATHML.pfx;mi" >
<!ENTITY % min.qname            "%MATHML.pfx;min" >
<!ENTITY % minus.qname          "%MATHML.pfx;minus" >
<!ENTITY % mlabeledtr.qname     "%MATHML.pfx;mlabeledtr" >
<!ENTITY % mmultiscripts.qname  "%MATHML.pfx;mmultiscripts" >
<!ENTITY % mn.qname             "%MATHML.pfx;mn" >
<!ENTITY % mo.qname             "%MATHML.pfx;mo" >
<!ENTITY % mode.qname           "%MATHML.pfx;mode" >
<!ENTITY % moment.qname         "%MATHML.pfx;moment" >
<!ENTITY % momentabout.qname    "%MATHML.pfx;momentabout" >
<!ENTITY % mover.qname          "%MATHML.pfx;mover" >
<!ENTITY % mpadded.qname        "%MATHML.pfx;mpadded" >
<!ENTITY % mphantom.qname       "%MATHML.pfx;mphantom" >
<!ENTITY % mprescripts.qname    "%MATHML.pfx;mprescripts" >
<!ENTITY % mroot.qname          "%MATHML.pfx;mroot" >
<!ENTITY % mrow.qname           "%MATHML.pfx;mrow" >
<!ENTITY % ms.qname             "%MATHML.pfx;ms" >
<!ENTITY % mspace.qname         "%MATHML.pfx;mspace" >
<!ENTITY % msqrt.qname          "%MATHML.pfx;msqrt" >
<!ENTITY % mstyle.qname         "%MATHML.pfx;mstyle" >
<!ENTITY % msub.qname           "%MATHML.pfx;msub" >
<!ENTITY % msubsup.qname        "%MATHML.pfx;msubsup" >
<!ENTITY % msup.qname           "%MATHML.pfx;msup" >
<!ENTITY % mtable.qname         "%MATHML.pfx;mtable" >
<!ENTITY % mtd.qname            "%MATHML.pfx;mtd" >
<!ENTITY % mtext.qname          "%MATHML.pfx;mtext" >
<!ENTITY % mtr.qname            "%MATHML.pfx;mtr" >
```

```
<!ENTITY % munder.qname         "%MATHML.pfx;munder" >
<!ENTITY % munderover.qname     "%MATHML.pfx;munderover" >
<!ENTITY % naturalnumbers.qname "%MATHML.pfx;naturalnumbers" >
<!ENTITY % neq.qname            "%MATHML.pfx;neq" >
<!ENTITY % none.qname           "%MATHML.pfx;none" >
<!ENTITY % not.qname            "%MATHML.pfx;not" >
<!ENTITY % notanumber.qname     "%MATHML.pfx;notanumber" >
<!ENTITY % notin.qname          "%MATHML.pfx;notin" >
<!ENTITY % notprsubset.qname    "%MATHML.pfx;notprsubset" >
<!ENTITY % notsubset.qname      "%MATHML.pfx;notsubset" >
<!ENTITY % or.qname             "%MATHML.pfx;or" >
<!ENTITY % otherwise.qname      "%MATHML.pfx;otherwise" >
<!ENTITY % outerproduct.qname   "%MATHML.pfx;outerproduct" >
<!ENTITY % partialdiff.qname    "%MATHML.pfx;partialdiff" >
<!ENTITY % pi.qname             "%MATHML.pfx;pi" >
<!ENTITY % piece.qname          "%MATHML.pfx;piece" >
<!ENTITY % piecewise.qname      "%MATHML.pfx;piecewise" >
<!ENTITY % plus.qname           "%MATHML.pfx;plus" >
<!ENTITY % power.qname          "%MATHML.pfx;power" >
<!ENTITY % primes.qname         "%MATHML.pfx;primes" >
<!ENTITY % product.qname        "%MATHML.pfx;product" >
<!ENTITY % prsubset.qname       "%MATHML.pfx;prsubset" >
<!ENTITY % quotient.qname       "%MATHML.pfx;quotient" >
<!ENTITY % rationals.qname      "%MATHML.pfx;rationals" >
<!ENTITY % real.qname           "%MATHML.pfx;real" >
<!ENTITY % reals.qname          "%MATHML.pfx;reals" >
<!ENTITY % reln.qname           "%MATHML.pfx;reln" >
<!ENTITY % rem.qname            "%MATHML.pfx;rem" >
<!ENTITY % root.qname           "%MATHML.pfx;root" >
<!ENTITY % scalarproduct.qname  "%MATHML.pfx;scalarproduct" >
<!ENTITY % sdev.qname           "%MATHML.pfx;sdev" >
<!ENTITY % sec.qname            "%MATHML.pfx;sec" >
<!ENTITY % sech.qname           "%MATHML.pfx;sech" >
<!ENTITY % selector.qname       "%MATHML.pfx;selector" >
<!ENTITY % semantics.qname      "%MATHML.pfx;semantics" >
<!ENTITY % sep.qname            "%MATHML.pfx;sep" >
<!ENTITY % set.qname            "%MATHML.pfx;set" >
<!ENTITY % setdiff.qname        "%MATHML.pfx;setdiff" >
<!ENTITY % sin.qname            "%MATHML.pfx;sin" >
<!ENTITY % sinh.qname           "%MATHML.pfx;sinh" >
<!ENTITY % subset.qname         "%MATHML.pfx;subset" >
<!ENTITY % sum.qname            "%MATHML.pfx;sum" >
<!ENTITY % tan.qname            "%MATHML.pfx;tan" >
<!ENTITY % tanh.qname           "%MATHML.pfx;tanh" >
<!ENTITY % tendsto.qname        "%MATHML.pfx;tendsto" >
<!ENTITY % times.qname          "%MATHML.pfx;times" >
<!ENTITY % transpose.qname      "%MATHML.pfx;transpose" >
<!ENTITY % true.qname           "%MATHML.pfx;true" >
```

```
<!ENTITY % union.qname          "%MATHML.pfx;union" >
<!ENTITY % uplimit.qname        "%MATHML.pfx;uplimit" >
<!ENTITY % variance.qname       "%MATHML.pfx;variance" >
<!ENTITY % vector.qname         "%MATHML.pfx;vector" >
<!ENTITY % vectorproduct.qname  "%MATHML.pfx;vectorproduct" >
<!ENTITY % xor.qname            "%MATHML.pfx;xor" >


<!-- ignores subsequent instantiation of this module when
   used as external subset rather than module fragment.
   NOTE: Do not modify this parameter entity, otherwise
   a recursive parsing situation may result.
-->
<!ENTITY % mathml-qname.module "IGNORE" >

<!-- end of template-qname-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/mathml/mathml2/mathml2-qname-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Paint Attribute Module ..................................... -->
<!-- file: svg-paint-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-paint-attrib.mod,v 1.2 2002/04/20 18:07:43 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Paint Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-paint-attrib.mod"

   ..................................................................... -->

<!-- Paint Attribute

     fill, fill-rule, stroke, stroke-dasharray, stroke-dashoffset,
     stroke-linecap, stroke-linejoin, stroke-miterlimit, stroke-width, color,
     color-interpolation, color-rendering

   This module defines the Paint and Color attribute sets.
-->

<!-- a 'fill' or 'stroke' property/attribute value: <paint> -->
<!ENTITY % Paint.datatype "CDATA" >
```

```
<!-- 'stroke-dasharray' property/attribute value (e.g., 'none', list of <number>s) -->
<!ENTITY % StrokeDashArrayValue.datatype "CDATA" >

<!-- 'stroke-dashoffset' property/attribute value (e.g., 'none', <legnth>) -->
<!ENTITY % StrokeDashOffsetValue.datatype "CDATA" >

<!-- 'stroke-miterlimit' property/attribute value (e.g., <number>) -->
<!ENTITY % StrokeMiterLimitValue.datatype "CDATA" >

<!-- 'stroke-width' property/attribute value (e.g., <length>) -->
<!ENTITY % StrokeWidthValue.datatype "CDATA" >

<!ENTITY % SVG.fill.attrib
  "fill %Paint.datatype; #IMPLIED"
>

<!ENTITY % SVG.fill-rule.attrib
  "fill-rule %ClipFillRule.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke.attrib
  "stroke %Paint.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-dasharray.attrib
  "stroke-dasharray %StrokeDashArrayValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-dashoffset.attrib
  "stroke-dashoffset %StrokeDashOffsetValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-linecap.attrib
  "stroke-linecap ( butt | round | square | inherit ) #IMPLIED"
>

<!ENTITY % SVG.stroke-linejoin.attrib
  "stroke-linejoin ( miter | round | bevel | inherit ) #IMPLIED"
>

<!ENTITY % SVG.stroke-miterlimit.attrib
  "stroke-miterlimit %StrokeMiterLimitValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-width.attrib
  "stroke-width %StrokeWidthValue.datatype; #IMPLIED"
>
```

```
<!ENTITY % SVG.Paint.extra.attrib "" >

<!ENTITY % SVG.Paint.attrib
  "%SVG.fill.attrib;
   %SVG.fill-rule.attrib;
   %SVG.stroke.attrib;
   %SVG.stroke-dasharray.attrib;
   %SVG.stroke-dashoffset.attrib;
   %SVG.stroke-linecap.attrib;
   %SVG.stroke-linejoin.attrib;
   %SVG.stroke-miterlimit.attrib;
   %SVG.stroke-width.attrib;
   %SVG.Paint.extra.attrib;"
>

<!ENTITY % SVG.color.attrib
  "color %Color.datatype; #IMPLIED"
>

<!ENTITY % SVG.color-interpolation.attrib
  "color-interpolation ( auto | sRGB | linearRGB | inherit ) #IMPLIED"
>

<!ENTITY % SVG.color-rendering.attrib
  "color-rendering ( auto | optimizeSpeed | optimizeQuality | inherit )
                #IMPLIED"
>

<!ENTITY % SVG.Color.extra.attrib "" >

<!ENTITY % SVG.Color.attrib
  "%SVG.color.attrib;
   %SVG.color-interpolation.attrib;
   %SVG.color-rendering.attrib;
   %SVG.Color.extra.attrib;"
>

<!-- end of svg-paint-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-paint-attrib.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- XHTML Document Metainformation Module  ..... -->
<!-- file: xhtml-meta-2.mod
```

```
   This is XHTML-RDFa, modules to annotate XHTML family documents.
   Copyright 2007 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-meta-2.mod,v 1.1 2007/04/02 16:44:01 jigsaw Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Metainformation 2.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-meta-2.mod"

   Revisions:
   (none)
   ..... -->

<!-- Meta Information

      meta
      link

   This module declares the meta and link element types,
   used to provide declarative document metainformation.
-->

<!-- meta: Generic Metainformation ..... -->

<!ENTITY % meta.element  "INCLUDE" >
<![%meta.element;[
<!ENTITY % meta.content  "( #PCDATA | %Inline.mix; )*" >
<!ENTITY % meta.qname  "meta" >
<!ELEMENT %meta.qname;  %meta.content; >
<!-- end of meta.element -->]]>

<!ENTITY % meta.attlist  "INCLUDE" >
<![%meta.attlist;[
<!ATTLIST %meta.qname;
    %Common.attrib;
    http-equiv   NMTOKEN               #IMPLIED
    name         NMTOKEN               #IMPLIED
    scheme       CDATA                 #IMPLIED
>
<!-- end of meta.attlist -->]]>

<!-- link: Media-Independent Link ..... -->

<!ENTITY % link.element  "INCLUDE" >
<![%link.element;[
<!ENTITY % link.content  "( %link.qname; | %meta.qname; )*" >
<!ENTITY % link.qname  "link" >
```

```
<!ELEMENT %link.qname;  %link.content; >
<!-- end of link.element -->]]>

<!ENTITY % link.attlist  "INCLUDE" >
<![%link.attlist;[
<!ATTLIST %link.qname;
    %Common.attrib;
    charset      %Charset.datatype;       #IMPLIED
    type         %ContentType.datatype;   #IMPLIED
    media        %MediaDesc.datatype;     #IMPLIED
>
<!-- end of link.attlist -->]]>

<!-- end of xhtml-meta-2.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-meta-2.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Tiny Attribute Collection Module ............................. -->
<!-- file: svg11-tiny-attribs.mod

   This is SVG Tiny, a proper subset of SVG.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg11-tiny-attribs.mod,v 1.3 2002/11/13 20:14:31 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Tiny Attribute Collection//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11-tiny-attribs.mod"

   ..................................................................... -->

<!-- SVG 1.1 Tiny Attribute Collection

   This module defines the set of common attributes that can be present
   on many SVG elements.
-->

<!-- module: svg-conditional.mod ....................... -->

<!ENTITY % ExtensionList.datatype "CDATA" >
<!ENTITY % FeatureList.datatype "CDATA" >

<!ENTITY % SVG.Conditional.extra.attrib "" >
<!ENTITY % SVG.Conditional.attrib
```

```
  "requiredFeatures %FeatureList.datatype; #IMPLIED
   requiredExtensions %ExtensionList.datatype; #IMPLIED
   systemLanguage %LanguageCodes.datatype; #IMPLIED
   %SVG.Conditional.extra.attrib;"
>

<!-- module: svg-basic-text.mod ..... -->

<!ENTITY % FontFamilyValue.datatype "CDATA" >
<!ENTITY % FontSizeValue.datatype "CDATA" >

<!ENTITY % SVG.TextContent.extra.attrib "" >
<!ENTITY % SVG.TextContent.attrib
  "text-anchor ( start | middle | end | inherit ) #IMPLIED
   %SVG.TextContent.extra.attrib;"
>

<!ENTITY % SVG.Font.extra.attrib "" >
<!ENTITY % SVG.Font.attrib
  "font-family %FontFamilyValue.datatype; #IMPLIED
   font-size %FontSizeValue.datatype; #IMPLIED
   font-style ( normal | italic | oblique | inherit ) #IMPLIED
   font-weight ( normal | bold | bolder | lighter | 100 | 200 | 300 | 400 |
              500 | 600 | 700 | 800 | 900 | inherit ) #IMPLIED
   %SVG.Font.extra.attrib;"
>

<!-- end of svg11-tiny-attribs.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg11-tiny-attribs.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Core Attribute Module ..................................... -->
<!-- file: svg-core-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-core-attrib.mod,v 1.3 2002/04/28 13:50:23 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Core Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-core-attrib.mod"

   ..................................................................... -->
```

```
<!-- Core Attribute

     id, xml:base, xml:lang, xml:space

   This module defines the core set of attributes that can be present on
   any element.
-->

<!ENTITY % SVG.id.attrib
  "id ID #IMPLIED"
>

<!ENTITY % SVG.base.attrib
  "xml:base %URI.datatype; #IMPLIED"
>

<!ENTITY % SVG.lang.attrib
  "xml:lang %LanguageCode.datatype; #IMPLIED"
>

<!ENTITY % SVG.space.attrib
  "xml:space ( default | preserve ) #IMPLIED"
>

<!ENTITY % SVG.Core.extra.attrib "" >

<!ENTITY % SVG.Core.attrib
  "%SVG.id.attrib;
   %SVG.base.attrib;
   %SVG.lang.attrib;
   %SVG.space.attrib;
   %SVG.Core.extra.attrib;"
>

<!-- end of svg-core-attrib.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-core-attrib.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Inline Structural Module  ..................................... -->
<!-- file: xhtml-inlstruct-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
```

```
   Revision: $Id: xhtml-inlstruct-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Inline Structural 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-inlstruct-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Inline Structural

      br, span

   This module declares the elements and their attributes
   used to support inline-level structural markup.
-->

<!-- br: forced line break ..... -->

<!ENTITY % br.element  "INCLUDE" >
<![%br.element;[

<!ENTITY % br.content  "EMPTY" >
<!ENTITY % br.qname  "br" >
<!ELEMENT %br.qname;  %br.content; >

<!-- end of br.element -->]]>

<!ENTITY % br.attlist  "INCLUDE" >
<![%br.attlist;[
<!ATTLIST %br.qname;
    %Core.attrib;
>
<!-- end of br.attlist -->]]>

<!-- span: generic inline container ..... -->

<!ENTITY % span.element  "INCLUDE" >
<![%span.element;[
<!ENTITY % span.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % span.qname  "span" >
<!ELEMENT %span.qname;  %span.content; >
<!-- end of span.element -->]]>
```

```
<!ENTITY % span.attlist  "INCLUDE" >
<![%span.attlist;[
<!ATTLIST %span.qname;
    %Common.attrib;
>
<!-- end of span.attlist -->]]>

<!-- end of xhtml-inlstruct-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-inlstruct-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Embedded Object Module  ....................................... -->
<!-- file: xhtml-object-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-object-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Embedded Object 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-object-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Embedded Objects

      object

   This module declares the object element type and its attributes,
   used to embed external objects as part of XHTML pages. In the
   document, place param elements prior to the object elements
   that require their content.

   Note that use of this module requires instantiation of the
   Param Element Module prior to this module.
-->

<!-- object: Generic Embedded Object ..................... -->

<!ENTITY % object.element  "INCLUDE" >
<![%object.element;[
```

```
<!ENTITY % object.content
   "( #PCDATA | %Flow.mix; | %param.qname; )*"
>
<!ENTITY % object.qname  "object" >
<!ELEMENT %object.qname;  %object.content; >
<!-- end of object.element -->]]>

<!ENTITY % object.attlist  "INCLUDE" >
<![%object.attlist;[
<!ATTLIST %object.qname;
    %Common.attrib;
    declare      ( declare )              #IMPLIED
    classid      %URI.datatype;           #IMPLIED
    codebase     %URI.datatype;           #IMPLIED
    data         %URI.datatype;           #IMPLIED
    type         %ContentType.datatype;   #IMPLIED
    codetype     %ContentType.datatype;   #IMPLIED
    archive      %URIs.datatype;          #IMPLIED
    standby      %Text.datatype;          #IMPLIED
    height       %Length.datatype;        #IMPLIED
    width        %Length.datatype;        #IMPLIED
    name         CDATA                    #IMPLIED
    tabindex     %Number.datatype;        #IMPLIED
>
<!-- end of object.attlist -->]]>

<!-- end of xhtml-object-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-object-1.mod
No license file was found, but licenses were detected in source scan.

```
<!--
   File isoamsc.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.
```

```
   Entity names in this file are derived from files carrying the
   following notice:

   (C) International Organization for Standardization 1991
   Permission to copy in any form is granted for use with
   conforming SGML systems and applications as defined in
   ISO 8879, provided this notice is included in all copies.

-->


<!--

     Public identifier: -//W3C//ENTITIES Added Math Symbols: Delimiters//EN//XML
     System identifier: http://www.w3.org/2003/entities/2007/isoamsc.ent

   The public identifier should always be used verbatim.
   The system identifier may be changed to suit local requirements.

   Typical invocation:

     <!ENTITY % isoamsc PUBLIC
       "-//W3C//ENTITIES Added Math Symbols: Delimiters//EN//XML"
       "http://www.w3.org/2003/entities/2007/isoamsc.ent"
     >
     %isoamsc;

-->

<!ENTITY dlcorn         "&#x0231E;" ><!--BOTTOM LEFT CORNER -->
<!ENTITY drcorn         "&#x0231F;" ><!--BOTTOM RIGHT CORNER -->
<!ENTITY gtlPar         "&#x02995;" ><!--DOUBLE LEFT ARC GREATER-THAN BRACKET -->
<!ENTITY langd          "&#x02991;" ><!--LEFT ANGLE BRACKET WITH DOT -->
<!ENTITY lbrke          "&#x0298B;" ><!--LEFT SQUARE BRACKET WITH UNDERBAR -->
<!ENTITY lbrksld        "&#x0298F;" ><!--LEFT SQUARE BRACKET WITH TICK IN BOTTOM CORNER -->
<!ENTITY lbrkslu        "&#x0298D;" ><!--LEFT SQUARE BRACKET WITH TICK IN TOP CORNER -->
<!ENTITY lceil          "&#x02308;" ><!--LEFT CEILING -->
<!ENTITY lfloor         "&#x0230A;" ><!--LEFT FLOOR -->
<!ENTITY lmoust         "&#x023B0;" ><!--UPPER LEFT OR LOWER RIGHT CURLY BRACKET SECTION --
>
<!ENTITY lparlt         "&#x02993;" ><!--LEFT ARC LESS-THAN BRACKET -->
<!ENTITY ltrPar         "&#x02996;" ><!--DOUBLE RIGHT ARC LESS-THAN BRACKET -->
<!ENTITY rangd          "&#x02992;" ><!--RIGHT ANGLE BRACKET WITH DOT -->
<!ENTITY rbrke          "&#x0298C;" ><!--RIGHT SQUARE BRACKET WITH UNDERBAR -->
<!ENTITY rbrksld        "&#x0298E;" ><!--RIGHT SQUARE BRACKET WITH TICK IN BOTTOM CORNER --
>
<!ENTITY rbrkslu        "&#x02990;" ><!--RIGHT SQUARE BRACKET WITH TICK IN TOP CORNER -->
```

```
<!ENTITY rceil          "&#x02309;" ><!--RIGHT CEILING -->
<!ENTITY rfloor         "&#x0230B;" ><!--RIGHT FLOOR -->
<!ENTITY rmoust          "&#x023B1;" ><!--UPPER RIGHT OR LOWER LEFT CURLY BRACKET SECTION --
>
<!ENTITY rpargt         "&#x02994;" ><!--RIGHT ARC GREATER-THAN BRACKET -->
<!ENTITY ulcorn         "&#x0231C;" ><!--TOP LEFT CORNER -->
<!ENTITY urcorn          "&#x0231D;" ><!--TOP RIGHT CORNER -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isoamsc.ent
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Frames Module  ................................................. -->
<!-- file: xhtml-frames-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-frames-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Frames 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-frames-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Frames

      frameset, frame, noframes

   This module declares frame-related element types and attributes.
-->

<!ENTITY % frameset.qname  "frameset" >
<!ENTITY % frame.qname  "frame" >
<!ENTITY % noframes.qname  "noframes" >

<!-- comma-separated list of MultiLength -->
<!ENTITY % MultiLengths.datatype "CDATA" >

<!-- The content model for XHTML documents depends on whether
   the <head> is followed by a <frameset> or <body> element.
-->
```

```
<!ENTITY % frameset.element  "INCLUDE" >
<![%frameset.element;[
<!ENTITY % frameset.content
   "(( %frameset.qname; | %frame.qname; )+, %noframes.qname;? )" >
<!ELEMENT %frameset.qname;  %frameset.content; >
<!-- end of frameset.element -->]]>

<!ENTITY % frameset.attlist  "INCLUDE" >
<![%frameset.attlist;[
<!ATTLIST %frameset.qname;
    %Core.attrib;
    rows        %MultiLengths.datatype;  #IMPLIED
    cols        %MultiLengths.datatype;  #IMPLIED
>
<!-- end of frameset.attlist -->]]>
<![%xhtml-events.module;[
<!ATTLIST %frameset.qname;
    onload      %Script.datatype;       #IMPLIED
    onunload    %Script.datatype;       #IMPLIED
>
]]>

<!-- reserved frame names start with "_" otherwise starts with letter -->

<!ENTITY % frame.element  "INCLUDE" >
<![%frame.element;[
<!ENTITY % frame.content  "EMPTY" >
<!ELEMENT %frame.qname;  %frame.content; >
<!-- end of frame.element -->]]>

<!ENTITY % frame.attlist  "INCLUDE" >
<![%frame.attlist;[
<!ATTLIST %frame.qname;
    %Core.attrib;
    longdesc    %URI.datatype;          #IMPLIED
    src         %URI.datatype;          #IMPLIED
    frameborder  ( 1 | 0 )              '1'
    marginwidth  %Pixels.datatype;      #IMPLIED
    marginheight %Pixels.datatype;      #IMPLIED
    noresize    ( noresize )            #IMPLIED
    scrolling   ( yes | no | auto )     'auto'
>
<!-- end of frame.attlist -->]]>

<!-- changes to other declarations ..... -->

<!-- redefine content model for html element,
   substituting frameset for body  -->
```

```
<!ENTITY % html.content
   "( %head.qname;, %frameset.qname; )"
>

<!-- alternate content container for non frame-based rendering -->

<!ENTITY % noframes.element  "INCLUDE" >
<![%noframes.element;[
<!ENTITY % noframes.content "( %body.qname; )">
<!ELEMENT %noframes.qname;  %noframes.content; >
<!-- end of noframes.element -->]]>

<!ENTITY % noframes.attlist  "INCLUDE" >
<![%noframes.attlist;[
<!ATTLIST %noframes.qname;
    %Common.attrib;
>
<!-- end of noframes.attlist -->]]>

<!-- end of xhtml-frames-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-frames-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Document Metainformation Module  ..... -->
<!-- file: xhtml-meta-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-meta-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Metainformation 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-meta-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Meta Information

      meta

   This module declares the meta element type and its attributes,
```

```
   used to provide declarative document metainformation.
-->

<!-- meta: Generic Metainformation ..... -->

<!ENTITY % meta.element  "INCLUDE" >
<![%meta.element;[
<!ENTITY % meta.content  "EMPTY" >
<!ENTITY % meta.qname  "meta" >
<!ELEMENT %meta.qname;  %meta.content; >
<!-- end of meta.element -->]]>

<!ENTITY % meta.attlist  "INCLUDE" >
<![%meta.attlist;[
<!ATTLIST %meta.qname;
    %XHTML.xmlns.attrib;
    %I18n.attrib;
    http-equiv   NMTOKEN               #IMPLIED
    name         NMTOKEN               #IMPLIED
    content      CDATA                 #REQUIRED
    scheme       CDATA                 #IMPLIED
>
<!-- end of meta.attlist -->]]>

<!-- end of xhtml-meta-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-meta-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Conditional Processing Module ..... -->
<!-- file: svg-conditional.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-conditional.mod,v 1.4 2002/11/14 15:11:02 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Conditional Processing//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-conditional.mod"

   ..................................................................... -->

<!-- Conditional Processing
```

```
      switch

   This module declares markup to provide support for conditional processing.
-->

<!-- extension list specification -->
<!ENTITY % ExtensionList.datatype "CDATA" >

<!-- feature list specification -->
<!ENTITY % FeatureList.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.switch.qname "switch" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Conditional.class ..... -->

<!ENTITY % SVG.Conditional.extra.class "" >

<!ENTITY % SVG.Conditional.class
   "| %SVG.switch.qname; %SVG.Conditional.extra.class;"
>

<!-- SVG.Conditional.attrib ..... -->
```

```
<!ENTITY % SVG.Conditional.extra.attrib "" >

<!ENTITY % SVG.Conditional.attrib
  "requiredFeatures %FeatureList.datatype; #IMPLIED
   requiredExtensions %ExtensionList.datatype; #IMPLIED
   systemLanguage %LanguageCodes.datatype; #IMPLIED
   %SVG.Conditional.extra.attrib;"
>

<!-- SVG.Presentation.attrib ..... -->

<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
   %SVG.Viewport.attrib;
   %SVG.Text.attrib;
   %SVG.TextContent.attrib;
   %SVG.Font.attrib;
   %SVG.Paint.attrib;
   %SVG.Color.attrib;
   %SVG.Opacity.attrib;
   %SVG.Graphics.attrib;
   %SVG.Marker.attrib;
   %SVG.ColorProfile.attrib;
   %SVG.Gradient.attrib;
   %SVG.Clip.attrib;
   %SVG.Mask.attrib;
   %SVG.Filter.attrib;
   %SVG.FilterColor.attrib;
   %SVG.Cursor.attrib;
   flood-color %SVGColor.datatype; #IMPLIED
   flood-opacity %OpacityValue.datatype; #IMPLIED
   lighting-color %SVGColor.datatype; #IMPLIED
   %SVG.Presentation.extra.attrib;"
>

<!-- switch: Switch Element ..... -->

<!ENTITY % SVG.switch.extra.content "" >

<!ENTITY % SVG.switch.element "INCLUDE" >
<![%SVG.switch.element;[
<!ENTITY % SVG.switch.content
  "(( %SVG.Description.class; )*, ( %SVG.svg.qname; | %SVG.g.qname;
    | %SVG.use.qname; | %SVG.text.qname; | %SVG.Animation.class;
      %SVG.Conditional.class; %SVG.Image.class; %SVG.Shape.class;
```

```
      %SVG.Hyperlink.class; %SVG.Extensibility.class;
      %SVG.switch.extra.content; )*)"
>
<!ELEMENT %SVG.switch.qname; %SVG.switch.content; >
<!-- end of SVG.switch.element --]]>

<!ENTITY % SVG.switch.attlist "INCLUDE" >
<![%SVG.switch.attlist;[
<!ATTLIST %SVG.switch.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.External.attrib;
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.switch.attlist --]]>

<!-- end of svg-conditional.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-conditional.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Basic Font Module ..... -->
<!-- file: svg-basic-font.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-basic-font.mod,v 1.5 2002/11/14 15:11:02 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Basic Font//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-font.mod"

   ..... -->

<!-- Basic Font

     font, font-face, glyph, missing-glyph, hkern, font-face-src,
     font-face-name

   This module declares markup to provide support for template.
-->
```

```
<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.font.qname "font" >
<!ENTITY % SVG.font-face.qname "font-face" >
<!ENTITY % SVG.glyph.qname "glyph" >
<!ENTITY % SVG.missing-glyph.qname "missing-glyph" >
<!ENTITY % SVG.hkern.qname "hkern" >
<!ENTITY % SVG.font-face-src.qname "font-face-src" >
<!ENTITY % SVG.font-face-name.qname "font-face-name" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLinkRequired.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Font.class ..... -->

<!ENTITY % SVG.Font.extra.class "" >

<!ENTITY % SVG.Font.class
   "| %SVG.font.qname; | %SVG.font-face.qname; %SVG.Font.extra.class;"
>

<!-- font: Font Element ..... -->

<!ENTITY % SVG.font.extra.content "" >

<!ENTITY % SVG.font.element "INCLUDE" >
```

```
<![%SVG.font.element;[
<!ENTITY % SVG.font.content
  "(( %SVG.Description.class; )*, %SVG.font-face.qname;,
     %SVG.missing-glyph.qname;, ( %SVG.glyph.qname; | %SVG.hkern.qname;
     %SVG.font.extra.content; )*)"
>
<!ELEMENT %SVG.font.qname; %SVG.font.content; >
<!-- end of SVG.font.element -->]]>

<!ENTITY % SVG.font.attlist "INCLUDE" >
<![%SVG.font.attlist;[
<!ATTLIST %SVG.font.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.External.attrib;
  horiz-origin-x %Number.datatype; #IMPLIED
  horiz-adv-x %Number.datatype; #REQUIRED
>
<!-- end of SVG.font.attlist -->]]>

<!-- font-face: Font Face Element ..... -->

<!ENTITY % SVG.font-face.extra.content "" >

<!ENTITY % SVG.font-face.element "INCLUDE" >
<![%SVG.font-face.element;[
<!ENTITY % SVG.font-face.content
  "(( %SVG.Description.class; )*, %SVG.font-face-src.qname;?
     %SVG.font-face.extra.content; )"
>
<!ELEMENT %SVG.font-face.qname; %SVG.font-face.content; >
<!-- end of SVG.font-face.element -->]]>

<!ENTITY % SVG.font-face.attlist "INCLUDE" >
<![%SVG.font-face.attlist;[
<!ATTLIST %SVG.font-face.qname;
  %SVG.Core.attrib;
  font-family CDATA #IMPLIED
  font-style CDATA #IMPLIED
  font-variant CDATA #IMPLIED
  font-weight CDATA #IMPLIED
  font-stretch CDATA #IMPLIED
  font-size CDATA #IMPLIED
  unicode-range CDATA #IMPLIED
  units-per-em %Number.datatype; #IMPLIED
  panose-1 CDATA #IMPLIED
  stemv %Number.datatype; #IMPLIED
  stemh %Number.datatype; #IMPLIED
```

```
  slope %Number.datatype; #IMPLIED
  cap-height %Number.datatype; #IMPLIED
  x-height %Number.datatype; #IMPLIED
  accent-height %Number.datatype; #IMPLIED
  ascent %Number.datatype; #IMPLIED
  descent %Number.datatype; #IMPLIED
  widths CDATA #IMPLIED
  bbox CDATA #IMPLIED
  ideographic %Number.datatype; #IMPLIED
  alphabetic %Number.datatype; #IMPLIED
  mathematical %Number.datatype; #IMPLIED
  hanging %Number.datatype; #IMPLIED
  underline-position %Number.datatype; #IMPLIED
  underline-thickness %Number.datatype; #IMPLIED
  strikethrough-position %Number.datatype; #IMPLIED
  strikethrough-thickness %Number.datatype; #IMPLIED
  overline-position %Number.datatype; #IMPLIED
  overline-thickness %Number.datatype; #IMPLIED
>
<!-- end of SVG.font-face.attlist -->]]>

<!-- glyph: Glyph Element ..... -->

<!ENTITY % SVG.glyph.extra.content "" >

<!ENTITY % SVG.glyph.element "INCLUDE" >
<![%SVG.glyph.element;[
<!ENTITY % SVG.glyph.content
  "( %SVG.Description.class; %SVG.glyph.extra.content; )*"
>
<!ELEMENT %SVG.glyph.qname; %SVG.glyph.content; >
<!-- end of SVG.glyph.element -->]]>

<!ENTITY % SVG.glyph.attlist "INCLUDE" >
<![%SVG.glyph.attlist;[
<!ATTLIST %SVG.glyph.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  unicode CDATA #IMPLIED
  glyph-name CDATA #IMPLIED
  d %PathData.datatype; #IMPLIED
  arabic-form CDATA #IMPLIED
  lang %LanguageCodes.datatype; #IMPLIED
  horiz-adv-x %Number.datatype; #IMPLIED
>
<!-- end of SVG.glyph.attlist -->]]>

<!-- missing-glyph: Missing Glyph Element ..... -->
```

```
<!ENTITY % SVG.missing-glyph.extra.content "" >

<!ENTITY % SVG.missing-glyph.element "INCLUDE" >
<![%SVG.missing-glyph.element;[
<!ENTITY % SVG.missing-glyph.content
  "( %SVG.Description.class; %SVG.missing-glyph.extra.content; )*"
>
<!ELEMENT %SVG.missing-glyph.qname; %SVG.missing-glyph.content; >
<!-- end of SVG.missing-glyph.element -->]]>

<!ENTITY % SVG.missing-glyph.attlist "INCLUDE" >
<![%SVG.missing-glyph.attlist;[
<!ATTLIST %SVG.missing-glyph.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  d %PathData.datatype; #IMPLIED
  horiz-adv-x %Number.datatype; #IMPLIED
>
<!-- end of SVG.missing-glyph.attlist -->]]>

<!-- hkern: Horizontal Kerning Element ..... -->

<!ENTITY % SVG.hkern.element "INCLUDE" >
<![%SVG.hkern.element;[
<!ENTITY % SVG.hkern.content "EMPTY" >
<!ELEMENT %SVG.hkern.qname; %SVG.hkern.content; >
<!-- end of SVG.hkern.element -->]]>

<!ENTITY % SVG.hkern.attlist "INCLUDE" >
<![%SVG.hkern.attlist;[
<!ATTLIST %SVG.hkern.qname;
  %SVG.Core.attrib;
  u1 CDATA #IMPLIED
  g1 CDATA #IMPLIED
  u2 CDATA #IMPLIED
  g2 CDATA #IMPLIED
  k %Number.datatype; #REQUIRED
>
<!-- end of SVG.hkern.attlist -->]]>

<!-- font-face-src: Font Face Source Element ..... -->

<!ENTITY % SVG.font-face-src.extra.content "" >

<!ENTITY % SVG.font-face-src.element "INCLUDE" >
<![%SVG.font-face-src.element;[
<!ENTITY % SVG.font-face-src.content
```

```
  "( %SVG.font-face-name.qname; %SVG.font-face-src.extra.content; )+"
>
<!ELEMENT %SVG.font-face-src.qname; %SVG.font-face-src.content; >
<!-- end of SVG.font-face-src.element -->]]>

<!ENTITY % SVG.font-face-src.attlist "INCLUDE" >
<![%SVG.font-face-src.attlist;[
<!ATTLIST %SVG.font-face-src.qname;
  %SVG.Core.attrib;
>
<!-- end of SVG.font-face-src.attlist -->]]>

<!-- font-face-name: Font Face Name Element ..... -->

<!ENTITY % SVG.font-face-name.element "INCLUDE" >
<![%SVG.font-face-name.element;[
<!ENTITY % SVG.font-face-name.content "EMPTY" >
<!ELEMENT %SVG.font-face-name.qname; %SVG.font-face-name.content; >
<!-- end of SVG.font-face-name.element -->]]>

<!ENTITY % SVG.font-face-name.attlist "INCLUDE" >
<![%SVG.font-face-name.attlist;[
<!ATTLIST %SVG.font-face-name.qname;
  %SVG.Core.attrib;
  name CDATA #IMPLIED
>
<!-- end of SVG.font-face-name.attlist -->]]>

<!-- end of svg-basic-font.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-basic-font.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML+RDFa Document Model Module  ..................................... -->
<!-- file: xhtml-rdfa-model-1.mod

   This is XHTML+RDFa.
   Copyright 1998-2008 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-rdfa-model-1.mod,v 1.4 2009/06/26 14:05:13 smccarro Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML+RDFa Document Model 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-model-1.mod"
```

```
    Revisions:
    (none)
    ..... -->

<!-- XHTML+RDFa Document Model

    This module describes the groupings of elements that make up
    common content models for XHTML elements.

    XHTML has three basic content models:

        %Inline.mix;  character-level elements
        %Block.mix;   block-like elements, eg., paragraphs and lists
        %Flow.mix;    any block or inline elements

    Any parameter entities declared in this module may be used
    to create element content models, but the above three are
    considered 'global' (insofar as that term applies here).

    The reserved word '#PCDATA' (indicating a text string) is now
    included explicitly with each element declaration that is
    declared as mixed content, as XML requires that this token
    occur first in a content model specification.
-->
<!-- Extending the Model

    While in some cases this module may need to be rewritten to
    accommodate changes to the document model, minor extensions
    may be accomplished by redeclaring any of the three *.extra;
    parameter entities to contain extension element types as follows:

        %Misc.extra;    whose parent may be any block or
                        inline element.

        %Inline.extra;  whose parent may be any inline element.

        %Block.extra;   whose parent may be any block element.

    If used, these parameter entities must be an OR-separated
    list beginning with an OR separator ("|"), eg., "| a | b | c"

    All block and inline *.class parameter entities not part
    of the *struct.class classes begin with "| " to allow for
    exclusion from mixes.
-->

<!-- .............  Optional Elements in head  ..................... -->
```

```
<!ENTITY % HeadOpts.mix
    "( %script.qname; | %style.qname; | %meta.qname;
     | %link.qname; | %object.qname; )*"
>

<!-- .....  Miscellaneous Elements  ..... -->

<!-- ins and del are used to denote editing changes
-->
<!ENTITY % Edit.class "| %ins.qname; | %del.qname;" >

<!-- script and noscript are used to contain scripts
   and alternative content
-->
<!ENTITY % Script.class "| %script.qname; | %noscript.qname;" >

<!ENTITY % Misc.extra "" >

<!-- These elements are neither block nor inline, and can
   essentially be used anywhere in the document body.
-->
<!ENTITY % Misc.class
    "%Edit.class;
     %Script.class;
     %Misc.extra;"
>

<!-- .....  Inline Elements  ..... -->

<!ENTITY % InlStruct.class "%br.qname; | %span.qname;" >

<!ENTITY % InlPhras.class
    "| %em.qname; | %strong.qname; | %dfn.qname; | %code.qname;
     | %samp.qname; | %kbd.qname; | %var.qname; | %cite.qname;
     | %abbr.qname; | %acronym.qname; | %q.qname;" >

<!ENTITY % InlPres.class
    "| %tt.qname; | %i.qname; | %b.qname; | %big.qname;
     | %small.qname; | %sub.qname; | %sup.qname;" >

<!ENTITY % I18n.class "| %bdo.qname;" >

<!ENTITY % Anchor.class "| %a.qname;" >

<!ENTITY % InlSpecial.class
    "| %img.qname; | %map.qname;
     | %object.qname;" >
```

```
<!ENTITY % InlForm.class
   "| %input.qname; | %select.qname; | %textarea.qname;
    | %label.qname; | %button.qname;" >

<!ENTITY % Inline.extra "" >

<!ENTITY % Ruby.class "| %ruby.qname;" >

<!-- %Inline.class; includes all inline elements,
   used as a component in mixes
-->
<!ENTITY % Inline.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %I18n.class;
    %Anchor.class;
    %InlSpecial.class;
    %InlForm.class;
    %Ruby.class;
    %Inline.extra;"
>

<!-- %InlNoRuby.class; includes all inline elements
   except ruby, used as a component in mixes
-->
<!ENTITY % InlNoRuby.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %I18n.class;
    %Anchor.class;
    %InlSpecial.class;
    %InlForm.class;
    %Inline.extra;"
>

<!-- %NoRuby.content; includes all inlines except ruby
-->
<!ENTITY % NoRuby.content
   "( #PCDATA
    | %InlNoRuby.class;
    %Misc.class; )*"
>

<!-- %InlNoAnchor.class; includes all non-anchor inlines,
   used as a component in mixes
-->
```

```
<!ENTITY % InlNoAnchor.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %I18n.class;
    %InlSpecial.class;
    %InlForm.class;
    %Ruby.class;
    %Inline.extra;"
>

<!-- %InlNoAnchor.mix; includes all non-anchor inlines
-->
<!ENTITY % InlNoAnchor.mix
   "%InlNoAnchor.class;
    %Misc.class;"
>

<!-- %Inline.mix; includes all inline elements, including %Misc.class;
-->
<!ENTITY % Inline.mix
   "%Inline.class;
    %Misc.class;"
>

<!-- .....................  Block Elements  ..................... -->

<!-- In the HTML 4.0 DTD, heading and list elements were included
   in the %block; parameter entity. The %Heading.class; and
   %List.class; parameter entities must now be included explicitly
   on element declarations where desired.
-->

<!ENTITY % Heading.class
   "%h1.qname; | %h2.qname; | %h3.qname;
    | %h4.qname; | %h5.qname; | %h6.qname;" >

<!ENTITY % List.class "%ul.qname; | %ol.qname; | %dl.qname;" >

<!ENTITY % Table.class "| %table.qname;" >

<!ENTITY % Form.class  "| %form.qname;" >

<!ENTITY % Fieldset.class  "| %fieldset.qname;" >

<!ENTITY % BlkStruct.class "%p.qname; | %div.qname;" >

<!ENTITY % BlkPhras.class
```

```
   "| %pre.qname; | %blockquote.qname; | %address.qname;" >

<!ENTITY % BlkPres.class "| %hr.qname; " >

<!ENTITY % BlkSpecial.class
   "%Table.class;
    %Form.class;
    %Fieldset.class;"
>

<!ENTITY % Block.extra "" >

<!-- %Block.class; includes all block elements,
   used as an component in mixes
-->
<!ENTITY % Block.class
   "%BlkStruct.class;
    %BlkPhras.class;
    %BlkPres.class;
    %BlkSpecial.class;
    %Block.extra;"
>

<!-- %Block.mix; includes all block elements plus %Misc.class;
-->
<!ENTITY % Block.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    %Misc.class;"
>

<!-- .....  All Content Elements  ..... -->

<!-- %Flow.mix; includes all text content, block and inline
-->
<!ENTITY % Flow.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    | %Inline.class;
    %Misc.class;"
>
<!-- end of xhtml-rdfa-model-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-rdfa-model-1.mod
```

No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML MetaAttributes Module  ..................................... -->
<!-- file: xhtml-metaAttributes-1.mod

   This is XHTML-RDFa, modules to annotate XHTML family documents.
   Copyright 2007-2008 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-metaAttributes-1.mod,v 1.6 2008/08/01 20:01:00 smccarro Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML MetaAttributes 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-metaAttributes-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!ENTITY % XHTML.global.attrs.prefixed "IGNORE" >

<!-- Placeholder Compact URI-related types -->
<!ENTITY % CURIE.datatype "CDATA" >
<!ENTITY % CURIEs.datatype "CDATA" >
<!ENTITY % SafeCURIE.datatype "CDATA" >
<!ENTITY % SafeCURIEs.datatype "CDATA" >
<!ENTITY % URIorSafeCURIE.datatype "CDATA" >
<!ENTITY % URIorSafeCURIEs.datatype "CDATA" >

<!-- Common Attributes

   This module declares a collection of meta-information related
   attributes.

   %NS.decl.attrib; is declared in the XHTML Qname module.

  This file also includes declarations of "global" versions of the
   attributes.  The global versions of the attributes are for use on
   elements in other namespaces.
-->

<!ENTITY % about.attrib
   "about       %URIorSafeCURIE.datatype;           #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.about.attrib
   "%XHTML.prefix;:about          %URIorSafeCURIE.datatype;       #IMPLIED"
```

```
>
]]>

<!ENTITY % typeof.attrib
   "typeof       %CURIEs.datatype;           #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.typeof.attrib
   "%XHTML.prefix;:typeof          %CURIEs.datatype;       #IMPLIED"
>
]]>

<!ENTITY % property.attrib
   "property       %CURIEs.datatype;           #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.property.attrib
   "%XHTML.prefix;:property          %CURIEs.datatype;       #IMPLIED"
>
]]>

<!ENTITY % resource.attrib
   "resource       %URIorSafeCURIE.datatype;           #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.resource.attrib
   "%XHTML.prefix;:resource          %URIorSafeCURIE.datatype;       #IMPLIED"
>
]]>

<!ENTITY % content.attrib
   "content       CDATA           #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.content.attrib
   "%XHTML.prefix;:content          CDATA       #IMPLIED"
>
]]>

<!ENTITY % datatype.attrib
   "datatype       %CURIE.datatype;           #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
```

```
<!ENTITY % XHTML.global.datatype.attrib
   "%XHTML.prefix;:datatype          %CURIE.datatype;       #IMPLIED"
>
]]>

<!ENTITY % rel.attrib
   "rel       %CURIEs.datatype;           #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.rel.attrib
   "%XHTML.prefix;:rel          %CURIEs.datatype;       #IMPLIED"
>
]]>

<!ENTITY % rev.attrib
   "rev       %CURIEs.datatype;           #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.rev.attrib
   "%XHTML.prefix;:rev          %CURIEs.datatype;       #IMPLIED"
>
]]>

<!ENTITY % Metainformation.extra.attrib "" >

<!ENTITY % Metainformation.attrib
   "%about.attrib;
    %content.attrib;
    %datatype.attrib;
    %typeof.attrib;
    %property.attrib;
    %rel.attrib;
    %resource.attrib;
    %rev.attrib;
    %Metainformation.extra.attrib;"
>

<!ENTITY % XHTML.global.metainformation.extra.attrib "" >

<![%XHTML.global.attrs.prefixed;[

<!ENTITY % XHTML.global.metainformation.attrib
   "%XHTML.global.about.attrib;
    %XHTML.global.content.attrib;
    %XHTML.global.datatype.attrib;
    %XHTML.global.typeof.attrib;
```

```
    %XHTML.global.property.attrib;
    %XHTML.global.rel.attrib;
    %XHTML.global.resource.attrib;
    %XHTML.global.rev.attrib;
    %XHTML.global.metainformation.extra.attrib;"
>
]]>

<!ENTITY % XHTML.global.metainformation.attrib "" >


<!-- end of xhtml-metaAttributes-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-metaAttributes-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Character Entities Module  ..................................... -->
<!-- file: xhtml-charent-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-charent-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Character Entities 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-charent-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Character Entities for XHTML

   This module declares the set of character entities for XHTML,
   including the Latin 1, Symbol and Special character collections.
-->

<!ENTITY % xhtml-lat1
  PUBLIC "-//W3C//ENTITIES Latin 1 for XHTML//EN"
        "xhtml-lat1.ent" >
<!ENTITY % xhtml-symbol
  PUBLIC "-//W3C//ENTITIES Symbols for XHTML//EN"
        "xhtml-symbol.ent" >
<!ENTITY % xhtml-special
```

```
  PUBLIC "-//W3C//ENTITIES Special for XHTML//EN"
        "xhtml-special.ent" >

%xhtml-lat1;
%xhtml-symbol;
%xhtml-special;

<!-- end of xhtml-charent-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-charent-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Image Module ................................................ -->
<!-- file: svg-image.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-image.mod,v 1.4 2002/11/14 15:11:03 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Image//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-image.mod"

   ..................................................................... -->

<!-- Image

     image

   This module declares markup to provide support for image.
-->

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.image.qname "image" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
```

```
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLinkEmbed.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Image.class ..................................... -->

<!ENTITY % SVG.Image.extra.class "" >

<!ENTITY % SVG.Image.class
  "| %SVG.image.qname; %SVG.Image.extra.class;"
>

<!-- image: Image Element ............................... -->

<!ENTITY % SVG.image.extra.content "" >

<!ENTITY % SVG.image.element "INCLUDE" >
<![%SVG.image.element;[
<!ENTITY % SVG.image.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class;
     %SVG.image.extra.content; )*)"
>
<!ELEMENT %SVG.image.qname; %SVG.image.content; >
<!-- end of SVG.image.element -->]]>

<!ENTITY % SVG.image.attlist "INCLUDE" >
<![%SVG.image.attlist;[
<!ATTLIST %SVG.image.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Viewport.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.ColorProfile.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.XLinkEmbed.attrib;
```

```
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #REQUIRED
  height %Length.datatype; #REQUIRED
  preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.image.attlist -->]]>

<!-- end of svg-image.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-image.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- XHTML Qname Module  ..................................................... -->
<!-- file: xhtml-qname-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-qname-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Qualified Names 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-qname-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- XHTML Qname (Qualified Name) Module

   This module is contained in two parts, labeled Section 'A' and 'B':

     Section A declares parameter entities to support namespace-
     qualified names, namespace declarations, and name prefixing
     for XHTML and extensions.

     Section B declares parameter entities used to provide
     namespace-qualified names for all XHTML element types:

       %applet.qname;   the xmlns-qualified name for <applet>
       %base.qname;     the xmlns-qualified name for <base>
       ...
```

```
   XHTML extensions would create a module similar to this one.
   Included in the XHTML distribution is a template module
   ('template-qname-1.mod') suitable for this purpose.
-->

<!-- Section A: XHTML XML Namespace Framework :::::::::::::::::::: -->

<!-- 1. Declare a %XHTML.prefixed; conditional section keyword, used
      to activate namespace prefixing. The default value should
      inherit '%NS.prefixed;' from the DTD driver, so that unless
      overridden, the default behaviour follows the overall DTD
      prefixing scheme.
-->
<!ENTITY % NS.prefixed "IGNORE" >
<!ENTITY % XHTML.prefixed "%NS.prefixed;" >

<!-- 2. Declare a parameter entity (eg., %XHTML.xmlns;) containing
      the URI reference used to identify the XHTML namespace:
-->
<!ENTITY % XHTML.xmlns  "http://www.w3.org/1999/xhtml" >

<!-- 3. Declare parameter entities (eg., %XHTML.prefix;) containing
      the default namespace prefix string(s) to use when prefixing
      is enabled. This may be overridden in the DTD driver or the
      internal subset of an document instance. If no default prefix
      is desired, this may be declared as an empty string.

   NOTE: As specified in [XMLNAMES], the namespace prefix serves
   as a proxy for the URI reference, and is not in itself significant.
-->
<!ENTITY % XHTML.prefix  "" >

<!-- 4. Declare parameter entities (eg., %XHTML.pfx;) containing the
      colonized prefix(es) (eg., '%XHTML.prefix;:') used when
      prefixing is active, an empty string when it is not.
-->
<![%XHTML.prefixed;[
<!ENTITY % XHTML.pfx  "%XHTML.prefix;:" >
]]>
<!ENTITY % XHTML.pfx  "" >

<!-- declare qualified name extensions here ............ -->
<!ENTITY % xhtml-qname-extra.mod "" >
%xhtml-qname-extra.mod;

<!-- 5. The parameter entity %XHTML.xmlns.extra.attrib; may be
      redeclared to contain any non-XHTML namespace declaration
```

```
      attributes for namespaces embedded in XHTML. The default
      is an empty string.  XLink should be included here if used
      in the DTD.
-->
<!ENTITY % XHTML.xmlns.extra.attrib "" >

<!-- The remainder of Section A is only followed in XHTML, not extensions. -->

<!-- Declare a parameter entity %NS.decl.attrib; containing
   all XML Namespace declarations used in the DTD, plus the
   xmlns declaration for XHTML, its form dependent on whether
   prefixing is active.
-->
<![%XHTML.prefixed;[
<!ENTITY % NS.decl.attrib
   "xmlns:%XHTML.prefix;  %URI.datatype;   #FIXED '%XHTML.xmlns;'
    %XHTML.xmlns.extra.attrib;"
>
]]>
<!ENTITY % NS.decl.attrib
   "%XHTML.xmlns.extra.attrib;"
>

<!-- This is a placeholder for future XLink support.
-->
<!ENTITY % XLINK.xmlns.attrib "" >

<!-- Declare a parameter entity %NS.decl.attrib; containing all
   XML namespace declaration attributes used by XHTML, including
   a default xmlns attribute when prefixing is inactive.
-->
<![%XHTML.prefixed;[
<!ENTITY % XHTML.xmlns.attrib
   "%NS.decl.attrib;
    %XLINK.xmlns.attrib;"
>
]]>
<!ENTITY % XHTML.xmlns.attrib
   "xmlns       %URI.datatype;          #FIXED '%XHTML.xmlns;'
    %XLINK.xmlns.attrib;"
>

<!-- placeholder for qualified name redeclarations -->
<!ENTITY % xhtml-qname.redecl "" >
%xhtml-qname.redecl;

<!-- Section B: XHTML Qualified Names :::::::::::::::::::::::::::: -->
```

```
<!-- 6. This section declares parameter entities used to provide
     namespace-qualified names for all XHTML element types.
-->

<!-- module:  xhtml-applet-1.mod -->
<!ENTITY % applet.qname  "%XHTML.pfx;applet" >

<!-- module:  xhtml-base-1.mod -->
<!ENTITY % base.qname    "%XHTML.pfx;base" >

<!-- module:  xhtml-bdo-1.mod -->
<!ENTITY % bdo.qname     "%XHTML.pfx;bdo" >

<!-- module:  xhtml-blkphras-1.mod -->
<!ENTITY % address.qname "%XHTML.pfx;address" >
<!ENTITY % blockquote.qname  "%XHTML.pfx;blockquote" >
<!ENTITY % pre.qname     "%XHTML.pfx;pre" >
<!ENTITY % h1.qname      "%XHTML.pfx;h1" >
<!ENTITY % h2.qname      "%XHTML.pfx;h2" >
<!ENTITY % h3.qname      "%XHTML.pfx;h3" >
<!ENTITY % h4.qname      "%XHTML.pfx;h4" >
<!ENTITY % h5.qname      "%XHTML.pfx;h5" >
<!ENTITY % h6.qname      "%XHTML.pfx;h6" >

<!-- module:  xhtml-blkpres-1.mod -->
<!ENTITY % hr.qname      "%XHTML.pfx;hr" >

<!-- module:  xhtml-blkstruct-1.mod -->
<!ENTITY % div.qname     "%XHTML.pfx;div" >
<!ENTITY % p.qname       "%XHTML.pfx;p" >

<!-- module:  xhtml-edit-1.mod -->
<!ENTITY % ins.qname     "%XHTML.pfx;ins" >
<!ENTITY % del.qname     "%XHTML.pfx;del" >

<!-- module:  xhtml-form-1.mod -->
<!ENTITY % form.qname    "%XHTML.pfx;form" >
<!ENTITY % label.qname   "%XHTML.pfx;label" >
<!ENTITY % input.qname   "%XHTML.pfx;input" >
<!ENTITY % select.qname  "%XHTML.pfx;select" >
<!ENTITY % optgroup.qname  "%XHTML.pfx;optgroup" >
<!ENTITY % option.qname  "%XHTML.pfx;option" >
<!ENTITY % textarea.qname  "%XHTML.pfx;textarea" >
<!ENTITY % fieldset.qname  "%XHTML.pfx;fieldset" >
<!ENTITY % legend.qname  "%XHTML.pfx;legend" >
<!ENTITY % button.qname  "%XHTML.pfx;button" >

<!-- module:  xhtml-hypertext-1.mod -->
```

```
<!ENTITY % a.qname       "%XHTML.pfx;a" >

<!-- module:  xhtml-image-1.mod -->
<!ENTITY % img.qname     "%XHTML.pfx;img" >

<!-- module:  xhtml-inlphras-1.mod -->
<!ENTITY % abbr.qname    "%XHTML.pfx;abbr" >
<!ENTITY % acronym.qname "%XHTML.pfx;acronym" >
<!ENTITY % cite.qname    "%XHTML.pfx;cite" >
<!ENTITY % code.qname    "%XHTML.pfx;code" >
<!ENTITY % dfn.qname     "%XHTML.pfx;dfn" >
<!ENTITY % em.qname      "%XHTML.pfx;em" >
<!ENTITY % kbd.qname     "%XHTML.pfx;kbd" >
<!ENTITY % q.qname       "%XHTML.pfx;q" >
<!ENTITY % samp.qname    "%XHTML.pfx;samp" >
<!ENTITY % strong.qname  "%XHTML.pfx;strong" >
<!ENTITY % var.qname     "%XHTML.pfx;var" >

<!-- module:  xhtml-inlpres-1.mod -->
<!ENTITY % b.qname       "%XHTML.pfx;b" >
<!ENTITY % big.qname     "%XHTML.pfx;big" >
<!ENTITY % i.qname       "%XHTML.pfx;i" >
<!ENTITY % small.qname   "%XHTML.pfx;small" >
<!ENTITY % sub.qname     "%XHTML.pfx;sub" >
<!ENTITY % sup.qname     "%XHTML.pfx;sup" >
<!ENTITY % tt.qname      "%XHTML.pfx;tt" >

<!-- module:  xhtml-inlstruct-1.mod -->
<!ENTITY % br.qname      "%XHTML.pfx;br" >
<!ENTITY % span.qname    "%XHTML.pfx;span" >

<!-- module:  xhtml-ismap-1.mod (also csismap, ssismap) -->
<!ENTITY % map.qname     "%XHTML.pfx;map" >
<!ENTITY % area.qname    "%XHTML.pfx;area" >

<!-- module:  xhtml-link-1.mod -->
<!ENTITY % link.qname    "%XHTML.pfx;link" >

<!-- module:  xhtml-list-1.mod -->
<!ENTITY % dl.qname      "%XHTML.pfx;dl" >
<!ENTITY % dt.qname      "%XHTML.pfx;dt" >
<!ENTITY % dd.qname      "%XHTML.pfx;dd" >
<!ENTITY % ol.qname      "%XHTML.pfx;ol" >
<!ENTITY % ul.qname      "%XHTML.pfx;ul" >
<!ENTITY % li.qname      "%XHTML.pfx;li" >

<!-- module:  xhtml-meta-1.mod -->
<!ENTITY % meta.qname    "%XHTML.pfx;meta" >
```

```
<!-- module:  xhtml-param-1.mod -->
<!ENTITY % param.qname   "%XHTML.pfx;param" >

<!-- module:  xhtml-object-1.mod -->
<!ENTITY % object.qname  "%XHTML.pfx;object" >

<!-- module:  xhtml-script-1.mod -->
<!ENTITY % script.qname  "%XHTML.pfx;script" >
<!ENTITY % noscript.qname  "%XHTML.pfx;noscript" >

<!-- module:  xhtml-struct-1.mod -->
<!ENTITY % html.qname    "%XHTML.pfx;html" >
<!ENTITY % head.qname    "%XHTML.pfx;head" >
<!ENTITY % title.qname   "%XHTML.pfx;title" >
<!ENTITY % body.qname    "%XHTML.pfx;body" >

<!-- module:  xhtml-style-1.mod -->
<!ENTITY % style.qname   "%XHTML.pfx;style" >

<!-- module:  xhtml-table-1.mod -->
<!ENTITY % table.qname   "%XHTML.pfx;table" >
<!ENTITY % caption.qname "%XHTML.pfx;caption" >
<!ENTITY % thead.qname   "%XHTML.pfx;thead" >
<!ENTITY % tfoot.qname   "%XHTML.pfx;tfoot" >
<!ENTITY % tbody.qname   "%XHTML.pfx;tbody" >
<!ENTITY % colgroup.qname  "%XHTML.pfx;colgroup" >
<!ENTITY % col.qname     "%XHTML.pfx;col" >
<!ENTITY % tr.qname      "%XHTML.pfx;tr" >
<!ENTITY % th.qname      "%XHTML.pfx;th" >
<!ENTITY % td.qname      "%XHTML.pfx;td" >


<!-- Provisional XHTML 2.0 Qualified Names  ..... -->

<!-- module:  xhtml-image-2.mod -->
<!ENTITY % alt.qname     "%XHTML.pfx;alt" >

<!-- end of xhtml-qname-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-qname-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Inline Presentation Module  ..... -->
<!-- file: xhtml-inlpres-1.mod
```

```
   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-inlpres-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Inline Presentation 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-inlpres-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Inline Presentational Elements

      b, big, i, small, sub, sup, tt

   This module declares the elements and their attributes used to
   support inline-level presentational markup.
-->

<!ENTITY % b.element  "INCLUDE" >
<![%b.element;[
<!ENTITY % b.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % b.qname  "b" >
<!ELEMENT %b.qname;  %b.content; >
<!-- end of b.element -->]]>

<!ENTITY % b.attlist  "INCLUDE" >
<![%b.attlist;[
<!ATTLIST %b.qname;
    %Common.attrib;
>
<!-- end of b.attlist -->]]>

<!ENTITY % big.element  "INCLUDE" >
<![%big.element;[
<!ENTITY % big.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % big.qname  "big" >
<!ELEMENT %big.qname;  %big.content; >
<!-- end of big.element -->]]>

<!ENTITY % big.attlist  "INCLUDE" >
```

```
<![%big.attlist;[
<!ATTLIST %big.qname;
    %Common.attrib;
>
<!-- end of big.attlist -->]]>

<!ENTITY % i.element  "INCLUDE" >
<![%i.element;[
<!ENTITY % i.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % i.qname  "i" >
<!ELEMENT %i.qname;  %i.content; >
<!-- end of i.element -->]]>

<!ENTITY % i.attlist  "INCLUDE" >
<![%i.attlist;[
<!ATTLIST %i.qname;
    %Common.attrib;
>
<!-- end of i.attlist -->]]>

<!ENTITY % small.element  "INCLUDE" >
<![%small.element;[
<!ENTITY % small.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % small.qname  "small" >
<!ELEMENT %small.qname;  %small.content; >
<!-- end of small.element -->]]>

<!ENTITY % small.attlist  "INCLUDE" >
<![%small.attlist;[
<!ATTLIST %small.qname;
    %Common.attrib;
>
<!-- end of small.attlist -->]]>

<!ENTITY % sub.element  "INCLUDE" >
<![%sub.element;[
<!ENTITY % sub.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % sub.qname  "sub" >
<!ELEMENT %sub.qname;  %sub.content; >
<!-- end of sub.element -->]]>

<!ENTITY % sub.attlist  "INCLUDE" >
```

```
<![%sub.attlist;[
<!ATTLIST %sub.qname;
    %Common.attrib;
>
<!-- end of sub.attlist -->]]>

<!ENTITY % sup.element  "INCLUDE" >
<![%sup.element;[
<!ENTITY % sup.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % sup.qname  "sup" >
<!ELEMENT %sup.qname;  %sup.content; >
<!-- end of sup.element -->]]>

<!ENTITY % sup.attlist  "INCLUDE" >
<![%sup.attlist;[
<!ATTLIST %sup.qname;
    %Common.attrib;
>
<!-- end of sup.attlist -->]]>

<!ENTITY % tt.element  "INCLUDE" >
<![%tt.element;[
<!ENTITY % tt.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % tt.qname  "tt" >
<!ELEMENT %tt.qname;  %tt.content; >
<!-- end of tt.element -->]]>

<!ENTITY % tt.attlist  "INCLUDE" >
<![%tt.attlist;[
<!ATTLIST %tt.qname;
    %Common.attrib;
>
<!-- end of tt.attlist -->]]>

<!-- end of xhtml-inlpres-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-inlpres-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Editing Elements Module  ..... -->
<!-- file: xhtml-edit-1.mod
```

```
   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-edit-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Editing Markup 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-edit-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Editing Elements

      ins, del

   This module declares element types and attributes used to indicate
   inserted and deleted content while editing a document.
-->

<!-- ins: Inserted Text  ..... -->

<!ENTITY % ins.element  "INCLUDE" >
<![%ins.element;[
<!ENTITY % ins.content
   "( #PCDATA | %Flow.mix; )*"
>
<!ENTITY % ins.qname  "ins" >
<!ELEMENT %ins.qname;  %ins.content; >
<!-- end of ins.element -->]]>

<!ENTITY % ins.attlist  "INCLUDE" >
<![%ins.attlist;[
<!ATTLIST %ins.qname;
    %Common.attrib;
    cite        %URI.datatype;          #IMPLIED
    datetime    %Datetime.datatype;     #IMPLIED
>
<!-- end of ins.attlist -->]]>

<!-- del: Deleted Text  ..... -->

<!ENTITY % del.element  "INCLUDE" >
<![%del.element;[
<!ENTITY % del.content
   "( #PCDATA | %Flow.mix; )*"
```

```
>
<!ENTITY % del.qname  "del" >
<!ELEMENT %del.qname;  %del.content; >
<!-- end of del.element -->]]>

<!ENTITY % del.attlist  "INCLUDE" >
<![%del.attlist;[
<!ATTLIST %del.qname;
    %Common.attrib;
    cite        %URI.datatype;          #IMPLIED
    datetime    %Datetime.datatype;     #IMPLIED
>
<!-- end of del.attlist -->]]>

<!-- end of xhtml-edit-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-edit-1.mod
No license file was found, but licenses were detected in source scan.

XHTML 1.1 plus MathML 2.0 plus SVG 1.1 DTD

This is a prototype extension of XHTML 1.1 incorporating MathML 2.0
and SVG 1.1.

Copyright 2002 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute this DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Editors: Murray M. Altheim <altheim@eng.sun.com> (XHTML modules)
David Carlisle <davidc@nag.co.uk> (MathML modules)
Jun Fujisawa <fujisawa.jun@canon.co.jp> (SVG modules)
Masayasu Ishikawa <mimasa@w3.org> (DTD driver)
Revision:
$Id: xhtml
2001 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).

All Rights Reserved.

Permission to use, copy, modify and distribute the XHTML DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Murray M. Altheim <altheim@eng.sun.com>
Revision: $Id: xhtml

2002 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the MathML 2.0 DTD and
its accompanying documentation for any purpose and without fee is
hereby granted in perpetuity, provided that the above copyright notice
and this paragraph appear in all copies. The copyright holders make
no representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Revision: $Id: xhtml

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml2/xhtml-math11-f-a.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Ruby Module ..... -->
<!-- file: xhtml-ruby-1.mod

This is XHTML, a reformulation of HTML as a modular XML application.
Copyright 1999-2001 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: xhtml-ruby-1.mod,v 1.1 2008/06/21 19:42:10 smccarro Exp $

This module is based on the W3C Ruby Annotation Specification:

http://www.w3.org/TR/ruby

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS XHTML Ruby 1.0//EN"
SYSTEM "http://www.w3.org/TR/ruby/xhtml-ruby-1.mod"

```
    ..... -->

<!-- Ruby Elements

      ruby, rbc, rtc, rb, rt, rp

   This module declares the elements and their attributes used to
   support ruby annotation markup.
-->

<!-- declare qualified element type names:
-->
<!ENTITY % ruby.qname  "ruby" >
<!ENTITY % rbc.qname  "rbc" >
<!ENTITY % rtc.qname  "rtc" >
<!ENTITY % rb.qname  "rb" >
<!ENTITY % rt.qname  "rt" >
<!ENTITY % rp.qname  "rp" >

<!-- rp fallback is included by default.
-->
<!ENTITY % Ruby.fallback "INCLUDE" >
<!ENTITY % Ruby.fallback.mandatory "IGNORE" >

<!-- Complex ruby is included by default; it may be
   overridden by other modules to ignore it.
-->
<!ENTITY % Ruby.complex "INCLUDE" >

<!-- Fragments for the content model of the ruby element -->
<![%Ruby.fallback;[
<![%Ruby.fallback.mandatory;[
<!ENTITY % Ruby.content.simple
   "( %rb.qname;, %rp.qname;, %rt.qname;, %rp.qname; )"
>
]]>
<!ENTITY % Ruby.content.simple
   "( %rb.qname;, ( %rt.qname; | ( %rp.qname;, %rt.qname;, %rp.qname; ) ) )"
>
]]>
<!ENTITY % Ruby.content.simple "( %rb.qname;, %rt.qname; )" >

<![%Ruby.complex;[
<!ENTITY % Ruby.content.complex
   "| ( %rbc.qname;, %rtc.qname;, %rtc.qname;? )"
>
]]>
```

```
<!ENTITY % Ruby.content.complex "" >

<!-- Content models of the rb and the rt elements are intended to
   allow other inline-level elements of its parent markup language,
   but it should not include ruby descendent elements. The following
   parameter entity %NoRuby.content; can be used to redefine
   those content models with minimum effort.  It's defined as
   '( #PCDATA )' by default.
-->
<!ENTITY % NoRuby.content "( #PCDATA )" >

<!-- one or more digits (NUMBER) -->
<!ENTITY % Number.datatype "CDATA" >

<!-- ruby element ..................................... -->

<!ENTITY % ruby.element  "INCLUDE" >
<![%ruby.element;[
<!ENTITY % ruby.content
   "( %Ruby.content.simple; %Ruby.content.complex; )"
>
<!ELEMENT %ruby.qname;  %ruby.content; >
<!-- end of ruby.element -->]]>

<![%Ruby.complex;[
<!-- rbc (ruby base component) element ..... -->

<!ENTITY % rbc.element  "INCLUDE" >
<![%rbc.element;[
<!ENTITY % rbc.content
   "(%rb.qname;)+"
>
<!ELEMENT %rbc.qname;  %rbc.content; >
<!-- end of rbc.element -->]]>

<!-- rtc (ruby text component) element ..... -->

<!ENTITY % rtc.element  "INCLUDE" >
<![%rtc.element;[
<!ENTITY % rtc.content
   "(%rt.qname;)+"
>
<!ELEMENT %rtc.qname;  %rtc.content; >
<!-- end of rtc.element -->]]>
]]>

<!-- rb (ruby base) element ........................... -->
```

```
<!ENTITY % rb.element  "INCLUDE" >
<![%rb.element;[
<!-- %rb.content; uses %NoRuby.content; as its content model,
   which is '( #PCDATA )' by default. It may be overridden
   by other modules to allow other inline-level elements
   of its parent markup language, but it should not include
   ruby descendent elements.
-->
<!ENTITY % rb.content "%NoRuby.content;" >
<!ELEMENT %rb.qname;  %rb.content; >
<!-- end of rb.element -->]]>

<!-- rt (ruby text) element ..... -->

<!ENTITY % rt.element  "INCLUDE" >
<![%rt.element;[
<!-- %rt.content; uses %NoRuby.content; as its content model,
   which is '( #PCDATA )' by default. It may be overridden
   by other modules to allow other inline-level elements
   of its parent markup language, but it should not include
   ruby descendent elements.
-->
<!ENTITY % rt.content "%NoRuby.content;" >

<!ELEMENT %rt.qname;  %rt.content; >
<!-- end of rt.element -->]]>

<!-- rbspan attribute is used for complex ruby only ..... -->
<![%Ruby.complex;[
<!ENTITY % rt.attlist  "INCLUDE" >
<![%rt.attlist;[
<!ATTLIST %rt.qname;
    rbspan        %Number.datatype;      "1"
>
<!-- end of rt.attlist -->]]>
]]>

<!-- rp (ruby parenthesis) element ..... -->

<![%Ruby.fallback;[
<!ENTITY % rp.element  "INCLUDE" >
<![%rp.element;[
<!ENTITY % rp.content
   "( #PCDATA )"
>
<!ELEMENT %rp.qname;  %rp.content; >
<!-- end of rp.element -->]]>
]]>
```

```
<!-- Ruby Common Attributes

   The following optional ATTLIST declarations provide an easy way
   to define common attributes for ruby elements.  These declarations
   are ignored by default.

   Ruby elements are intended to have common attributes of its
   parent markup language.  For example, if a markup language defines
   common attributes as a parameter entity %attrs;, you may add
   those attributes by just declaring the following parameter entities

      <!ENTITY % Ruby.common.attlists  "INCLUDE" >
      <!ENTITY % Ruby.common.attrib  "%attrs;" >

   before including the Ruby module.
-->

<!ENTITY % Ruby.common.attlists  "IGNORE" >
<![%Ruby.common.attlists;[
<!ENTITY % Ruby.common.attrib  "" >

<!-- common attributes for ruby ..... -->

<!ENTITY % Ruby.common.attlist  "INCLUDE" >
<![%Ruby.common.attlist;[
<!ATTLIST %ruby.qname;
    %Ruby.common.attrib;
>
<!-- end of Ruby.common.attlist -->]]>

<![%Ruby.complex;[
<!-- common attributes for rbc ..... -->

<!ENTITY % Rbc.common.attlist  "INCLUDE" >
<![%Rbc.common.attlist;[
<!ATTLIST %rbc.qname;
    %Ruby.common.attrib;
>
<!-- end of Rbc.common.attlist -->]]>

<!-- common attributes for rtc ..... -->

<!ENTITY % Rtc.common.attlist  "INCLUDE" >
<![%Rtc.common.attlist;[
<!ATTLIST %rtc.qname;
    %Ruby.common.attrib;
>
```

```
<!-- end of Rtc.common.attlist -->]]>
]]>

<!-- common attributes for rb ..... -->

<!ENTITY % Rb.common.attlist  "INCLUDE" >
<![%Rb.common.attlist;[
<!ATTLIST %rb.qname;
    %Ruby.common.attrib;
>
<!-- end of Rb.common.attlist -->]]>

<!-- common attributes for rt ..... -->

<!ENTITY % Rt.common.attlist  "INCLUDE" >
<![%Rt.common.attlist;[
<!ATTLIST %rt.qname;
    %Ruby.common.attrib;
>
<!-- end of Rt.common.attlist -->]]>

<![%Ruby.fallback;[
<!-- common attributes for rp ..... -->

<!ENTITY % Rp.common.attlist  "INCLUDE" >
<![%Rp.common.attlist;[
<!ATTLIST %rp.qname;
    %Ruby.common.attrib;
>
<!-- end of Rp.common.attlist -->]]>
]]>
]]>

<!-- end of xhtml-ruby-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-ruby-1.mod
No license file was found, but licenses were detected in source scan.

```
<!--
   File mmlalias.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
```

```
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

   References to the VARIANT SELECTOR 1 character (&#x0FE00;)
   should match the uses listed in Unicode Technical Report 25.

-->


<!--

     Public identifier: -//W3C//ENTITIES MathML Aliases//EN//XML
     System identifier: http://www.w3.org/2003/entities/2007/mmlalias.ent

   The public identifier should always be used verbatim.
   The system identifier may be changed to suit local requirements.

   Typical invocation:

     <!ENTITY % mmlalias PUBLIC
       "-//W3C//ENTITIES MathML Aliases//EN//XML"
       "http://www.w3.org/2003/entities/2007/mmlalias.ent"
     >
     %mmlalias;

-->

<!ENTITY angle           "&#x02220;" ><!--ANGLE -->
<!ENTITY ApplyFunction    "&#x02061;" ><!--FUNCTION APPLICATION -->
<!ENTITY approx          "&#x02248;" ><!--ALMOST EQUAL TO -->
<!ENTITY approxeq         "&#x0224A;" ><!--ALMOST EQUAL OR EQUAL TO -->
<!ENTITY Assign          "&#x02254;" ><!--COLON EQUALS -->
<!ENTITY backcong         "&#x0224C;" ><!--ALL EQUAL TO -->
<!ENTITY backepsilon      "&#x003F6;" ><!--GREEK REVERSED LUNATE EPSILON SYMBOL -->
<!ENTITY backprime        "&#x02035;" ><!--REVERSED PRIME -->
<!ENTITY backsim          "&#x0223D;" ><!--REVERSED TILDE -->
<!ENTITY backsimeq        "&#x022CD;" ><!--REVERSED TILDE EQUALS -->
<!ENTITY Backslash        "&#x02216;" ><!--SET MINUS -->
<!ENTITY barwedge         "&#x02305;" ><!--PROJECTIVE -->
<!ENTITY because          "&#x02235;" ><!--BECAUSE -->
<!ENTITY Because          "&#x02235;" ><!--BECAUSE -->
<!ENTITY Bernoullis       "&#x0212C;" ><!--SCRIPT CAPITAL B -->
<!ENTITY between          "&#x0226C;" ><!--BETWEEN -->
```

<!ENTITY bigcap "⋂" ><!--N-ARY INTERSECTION -->
<!ENTITY bigcirc "◯" ><!--LARGE CIRCLE -->
<!ENTITY bigcup "⋃" ><!--N-ARY UNION -->
<!ENTITY bigodot "⨀" ><!--N-ARY CIRCLED DOT OPERATOR -->
<!ENTITY bigoplus "⨁" ><!--N-ARY CIRCLED PLUS OPERATOR -->
<!ENTITY bigotimes "⨂" ><!--N-ARY CIRCLED TIMES OPERATOR -->
<!ENTITY bigsqcup "⨆" ><!--N-ARY SQUARE UNION OPERATOR -->
<!ENTITY bigstar "★" ><!--BLACK STAR -->
<!ENTITY bigtriangledown "▽" ><!--WHITE DOWN-POINTING TRIANGLE -->
<!ENTITY bigtriangleup "△" ><!--WHITE UP-POINTING TRIANGLE -->
<!ENTITY biguplus "⨄" ><!--N-ARY UNION OPERATOR WITH PLUS -->
<!ENTITY bigvee "⋁" ><!--N-ARY LOGICAL OR -->
<!ENTITY bigwedge "⋀" ><!--N-ARY LOGICAL AND -->
<!ENTITY bkarow "⤍" ><!--RIGHTWARDS DOUBLE DASH ARROW -->
<!ENTITY blacklozenge "⧫" ><!--BLACK LOZENGE -->
<!ENTITY blacksquare "▪" ><!--BLACK SMALL SQUARE -->
<!ENTITY blacktriangle "▴" ><!--BLACK UP-POINTING SMALL TRIANGLE -->
<!ENTITY blacktriangledown "▾" ><!--BLACK DOWN-POINTING SMALL TRIANGLE -->
<!ENTITY blacktriangleleft "◂" ><!--BLACK LEFT-POINTING SMALL TRIANGLE -->
<!ENTITY blacktriangleright "▸" ><!--BLACK RIGHT-POINTING SMALL TRIANGLE -->
<!ENTITY bot "⊥" ><!--UP TACK -->
<!ENTITY boxminus "⊟" ><!--SQUARED MINUS -->
<!ENTITY boxplus "⊞" ><!--SQUARED PLUS -->
<!ENTITY boxtimes "⊠" ><!--SQUARED TIMES -->
<!ENTITY Breve "˘" ><!--BREVE -->
<!ENTITY bullet "•" ><!--BULLET -->
<!ENTITY bumpeq "≏" ><!--DIFFERENCE BETWEEN -->
<!ENTITY Bumpeq "≎" ><!--GEOMETRICALLY EQUIVALENT TO -->
<!ENTITY CapitalDifferentialD "ⅅ" ><!--DOUBLE-STRUCK ITALIC CAPITAL D -->
<!ENTITY Cayleys "ℭ" ><!--BLACK-LETTER CAPITAL C -->
<!ENTITY Cedilla "¸" ><!--CEDILLA -->
<!ENTITY centerdot "·" ><!--MIDDLE DOT -->
<!ENTITY CenterDot "·" ><!--MIDDLE DOT -->
<!ENTITY checkmark "✓" ><!--CHECK MARK -->
<!ENTITY circeq "≗" ><!--RING EQUAL TO -->
<!ENTITY circlearrowleft "↺" ><!--ANTICLOCKWISE OPEN CIRCLE ARROW -->
<!ENTITY circlearrowright "↻" ><!--CLOCKWISE OPEN CIRCLE ARROW -->
<!ENTITY circledast "⊛" ><!--CIRCLED ASTERISK OPERATOR -->
<!ENTITY circledcirc "⊚" ><!--CIRCLED RING OPERATOR -->
<!ENTITY circleddash "⊝" ><!--CIRCLED DASH -->
<!ENTITY CircleDot "⊙" ><!--CIRCLED DOT OPERATOR -->
<!ENTITY circledR "®" ><!--REGISTERED SIGN -->
<!ENTITY circledS "Ⓢ" ><!--CIRCLED LATIN CAPITAL LETTER S -->
<!ENTITY CircleMinus "⊖" ><!--CIRCLED MINUS -->
<!ENTITY CirclePlus "⊕" ><!--CIRCLED PLUS -->
<!ENTITY CircleTimes "⊗" ><!--CIRCLED TIMES -->
<!ENTITY ClockwiseContourIntegral "∲" ><!--CLOCKWISE CONTOUR INTEGRAL -->
<!ENTITY CloseCurlyDoubleQuote "”" ><!--RIGHT DOUBLE QUOTATION MARK -->

<!ENTITY CloseCurlyQuote "’" ><!--RIGHT SINGLE QUOTATION MARK -->
<!ENTITY clubsuit "♣" ><!--BLACK CLUB SUIT -->
<!ENTITY coloneq "≔" ><!--COLON EQUALS -->
<!ENTITY complement "∁" ><!--COMPLEMENT -->
<!ENTITY complexes "ℂ" ><!--DOUBLE-STRUCK CAPITAL C -->
<!ENTITY Congruent "≡" ><!--IDENTICAL TO -->
<!ENTITY ContourIntegral "∮" ><!--CONTOUR INTEGRAL -->
<!ENTITY Coproduct "∐" ><!--N-ARY COPRODUCT -->
<!ENTITY CounterClockwiseContourIntegral "∳" ><!--ANTICLOCKWISE CONTOUR INTEGRAL -->
<!ENTITY CupCap "≍" ><!--EQUIVALENT TO -->
<!ENTITY curlyeqprec "⋞" ><!--EQUAL TO OR PRECEDES -->
<!ENTITY curlyeqsucc "⋟" ><!--EQUAL TO OR SUCCEEDS -->
<!ENTITY curlyvee "⋎" ><!--CURLY LOGICAL OR -->
<!ENTITY curlywedge "⋏" ><!--CURLY LOGICAL AND -->
<!ENTITY curvearrowleft "↶" ><!--ANTICLOCKWISE TOP SEMICIRCLE ARROW -->
<!ENTITY curvearrowright "↷" ><!--CLOCKWISE TOP SEMICIRCLE ARROW -->
<!ENTITY dbkarow "⤏" ><!--RIGHTWARDS TRIPLE DASH ARROW -->
<!ENTITY ddagger "‡" ><!--DOUBLE DAGGER -->
<!ENTITY ddotseq "⩷" ><!--EQUALS SIGN WITH TWO DOTS ABOVE AND TWO DOTS BELOW -->
<!ENTITY Del "∇" ><!--NABLA -->
<!ENTITY DiacriticalAcute "´" ><!--ACUTE ACCENT -->
<!ENTITY DiacriticalDot "˙" ><!--DOT ABOVE -->
<!ENTITY DiacriticalDoubleAcute "˝" ><!--DOUBLE ACUTE ACCENT -->
<!ENTITY DiacriticalGrave "`" ><!--GRAVE ACCENT -->
<!ENTITY DiacriticalTilde "˜" ><!--SMALL TILDE -->
<!ENTITY diamond "⋄" ><!--DIAMOND OPERATOR -->
<!ENTITY Diamond "⋄" ><!--DIAMOND OPERATOR -->
<!ENTITY diamondsuit "♦" ><!--BLACK DIAMOND SUIT -->
<!ENTITY DifferentialD "ⅆ" ><!--DOUBLE-STRUCK ITALIC SMALL D -->
<!ENTITY digamma "ϝ" ><!--GREEK SMALL LETTER DIGAMMA -->
<!ENTITY div "÷" ><!--DIVISION SIGN -->
<!ENTITY divideontimes "⋇" ><!--DIVISION TIMES -->
<!ENTITY doteq "≐" ><!--APPROACHES THE LIMIT -->
<!ENTITY doteqdot "≑" ><!--GEOMETRICALLY EQUAL TO -->
<!ENTITY DotEqual "≐" ><!--APPROACHES THE LIMIT -->
<!ENTITY dotminus "∸" ><!--DOT MINUS -->
<!ENTITY dotplus "∔" ><!--DOT PLUS -->
<!ENTITY dotsquare "⊡" ><!--SQUARED DOT OPERATOR -->
<!ENTITY doublebarwedge "⌆" ><!--PERSPECTIVE -->
<!ENTITY DoubleContourIntegral "∯" ><!--SURFACE INTEGRAL -->
<!ENTITY DoubleDot "¨" ><!--DIAERESIS -->
<!ENTITY DoubleDownArrow "⇓" ><!--DOWNWARDS DOUBLE ARROW -->
<!ENTITY DoubleLeftArrow "⇐" ><!--LEFTWARDS DOUBLE ARROW -->
<!ENTITY DoubleLeftRightArrow "⇔" ><!--LEFT RIGHT DOUBLE ARROW -->
<!ENTITY DoubleLeftTee "⫤" ><!--VERTICAL BAR DOUBLE LEFT TURNSTILE -->
<!ENTITY DoubleLongLeftArrow "⟸" ><!--LONG LEFTWARDS DOUBLE ARROW -->
<!ENTITY DoubleLongLeftRightArrow "⟺" ><!--LONG LEFT RIGHT DOUBLE ARROW -->

<!ENTITY DoubleLongRightArrow "⟹" ><!--LONG RIGHTWARDS DOUBLE ARROW -->
<!ENTITY DoubleRightArrow "⇒" ><!--RIGHTWARDS DOUBLE ARROW -->
<!ENTITY DoubleRightTee "⊨" ><!--TRUE -->
<!ENTITY DoubleUpArrow "⇑" ><!--UPWARDS DOUBLE ARROW -->
<!ENTITY DoubleUpDownArrow "⇕" ><!--UP DOWN DOUBLE ARROW -->
<!ENTITY DoubleVerticalBar "∥" ><!--PARALLEL TO -->
<!ENTITY downarrow "↓" ><!--DOWNWARDS ARROW -->
<!ENTITY Downarrow "⇓" ><!--DOWNWARDS DOUBLE ARROW -->
<!ENTITY DownArrow "↓" ><!--DOWNWARDS ARROW -->
<!ENTITY DownArrowUpArrow "⇵" ><!--DOWNWARDS ARROW LEFTWARDS OF UPWARDS ARROW -->
<!ENTITY downdownarrows "⇊" ><!--DOWNWARDS PAIRED ARROWS -->
<!ENTITY downharpoonleft "⇃" ><!--DOWNWARDS HARPOON WITH BARB LEFTWARDS -->
<!ENTITY downharpoonright "⇂" ><!--DOWNWARDS HARPOON WITH BARB RIGHTWARDS -->
<!ENTITY DownLeftVector "↽" ><!--LEFTWARDS HARPOON WITH BARB DOWNWARDS -->
<!ENTITY DownRightVector "⇁" ><!--RIGHTWARDS HARPOON WITH BARB DOWNWARDS -->
<!ENTITY DownTee "⊤" ><!--DOWN TACK -->
<!ENTITY DownTeeArrow "↧" ><!--DOWNWARDS ARROW FROM BAR -->
<!ENTITY drbkarow "⤐" ><!--RIGHTWARDS TWO-HEADED TRIPLE DASH ARROW -->
<!ENTITY Element "∈" ><!--ELEMENT OF -->
<!ENTITY emptyset "∅" ><!--EMPTY SET -->
<!ENTITY eqcirc "≖" ><!--RING IN EQUAL TO -->
<!ENTITY eqcolon "≕" ><!--EQUALS COLON -->
<!ENTITY eqsim "≂" ><!--MINUS TILDE -->
<!ENTITY eqslantgtr "⪖" ><!--SLANTED EQUAL TO OR GREATER-THAN -->
<!ENTITY eqslantless "⪕" ><!--SLANTED EQUAL TO OR LESS-THAN -->
<!ENTITY EqualTilde "≂" ><!--MINUS TILDE -->
<!ENTITY Equilibrium "⇌" ><!--RIGHTWARDS HARPOON OVER LEFTWARDS HARPOON -->
<!ENTITY Exists "∃" ><!--THERE EXISTS -->
<!ENTITY expectation "ℰ" ><!--SCRIPT CAPITAL E -->
<!ENTITY exponentiale "ⅇ" ><!--DOUBLE-STRUCK ITALIC SMALL E -->
<!ENTITY ExponentialE "ⅇ" ><!--DOUBLE-STRUCK ITALIC SMALL E -->
<!ENTITY fallingdotseq "≒" ><!--APPROXIMATELY EQUAL TO OR THE IMAGE OF -->
<!ENTITY ForAll "∀" ><!--FOR ALL -->
<!ENTITY Fouriertrf "ℱ" ><!--SCRIPT CAPITAL F -->
<!ENTITY geq "≥" ><!--GREATER-THAN OR EQUAL TO -->
<!ENTITY geqq "≧" ><!--GREATER-THAN OVER EQUAL TO -->
<!ENTITY geqslant "⩾" ><!--GREATER-THAN OR SLANTED EQUAL TO -->
<!ENTITY gg "≫" ><!--MUCH GREATER-THAN -->
<!ENTITY ggg "⋙" ><!--VERY MUCH GREATER-THAN -->
<!ENTITY gnapprox "⪊" ><!--GREATER-THAN AND NOT APPROXIMATE -->
<!ENTITY gneq "⪈" ><!--GREATER-THAN AND SINGLE-LINE NOT EQUAL TO -->
<!ENTITY gneqq "≩" ><!--GREATER-THAN BUT NOT EQUAL TO -->
<!ENTITY GreaterEqual "≥" ><!--GREATER-THAN OR EQUAL TO -->
<!ENTITY GreaterEqualLess "⋛" ><!--GREATER-THAN EQUAL TO OR LESS-THAN -->
<!ENTITY GreaterFullEqual "≧" ><!--GREATER-THAN OVER EQUAL TO -->
<!ENTITY GreaterLess "≷" ><!--GREATER-THAN OR LESS-THAN -->
<!ENTITY GreaterSlantEqual "⩾" ><!--GREATER-THAN OR SLANTED EQUAL TO -->

<!ENTITY GreaterTilde "≳" ><!--GREATER-THAN OR EQUIVALENT TO -->
<!ENTITY gtrapprox "⪆" ><!--GREATER-THAN OR APPROXIMATE -->
<!ENTITY gtrdot "⋗" ><!--GREATER-THAN WITH DOT -->
<!ENTITY gtreqless "⋛" ><!--GREATER-THAN EQUAL TO OR LESS-THAN -->
<!ENTITY gtreqqless "⪌" ><!--GREATER-THAN ABOVE DOUBLE-LINE EQUAL ABOVE LESS-THAN -->
<!ENTITY gtrless "≷" ><!--GREATER-THAN OR LESS-THAN -->
<!ENTITY gtrsim "≳" ><!--GREATER-THAN OR EQUIVALENT TO -->
<!ENTITY gvertneqq "≩︀" ><!--GREATER-THAN BUT NOT EQUAL TO - with vertical stroke -->
<!ENTITY Hacek "ˇ" ><!--CARON -->
<!ENTITY hbar "ℏ" ><!--PLANCK CONSTANT OVER TWO PI -->
<!ENTITY heartsuit "♥" ><!--BLACK HEART SUIT -->
<!ENTITY HilbertSpace "ℋ" ><!--SCRIPT CAPITAL H -->
<!ENTITY hksearow "⤥" ><!--SOUTH EAST ARROW WITH HOOK -->
<!ENTITY hkswarow "⤦" ><!--SOUTH WEST ARROW WITH HOOK -->
<!ENTITY hookleftarrow "↩" ><!--LEFTWARDS ARROW WITH HOOK -->
<!ENTITY hookrightarrow "↪" ><!--RIGHTWARDS ARROW WITH HOOK -->
<!ENTITY hslash "ℏ" ><!--PLANCK CONSTANT OVER TWO PI -->
<!ENTITY HumpDownHump "≎" ><!--GEOMETRICALLY EQUIVALENT TO -->
<!ENTITY HumpEqual "≏" ><!--DIFFERENCE BETWEEN -->
<!ENTITY iiiint "⨌" ><!--QUADRUPLE INTEGRAL OPERATOR -->
<!ENTITY iiint "∭" ><!--TRIPLE INTEGRAL -->
<!ENTITY Im "ℑ" ><!--BLACK-LETTER CAPITAL I -->
<!ENTITY ImaginaryI "ⅈ" ><!--DOUBLE-STRUCK ITALIC SMALL I -->
<!ENTITY imagline "ℐ" ><!--SCRIPT CAPITAL I -->
<!ENTITY imagpart "ℑ" ><!--BLACK-LETTER CAPITAL I -->
<!ENTITY Implies "⇒" ><!--RIGHTWARDS DOUBLE ARROW -->
<!ENTITY in "∈" ><!--ELEMENT OF -->
<!ENTITY integers "ℤ" ><!--DOUBLE-STRUCK CAPITAL Z -->
<!ENTITY Integral "∫" ><!--INTEGRAL -->
<!ENTITY intercal "⊺" ><!--INTERCALATE -->
<!ENTITY Intersection "⋂" ><!--N-ARY INTERSECTION -->
<!ENTITY intprod "⨼" ><!--INTERIOR PRODUCT -->
<!ENTITY InvisibleComma "⁣" ><!--INVISIBLE SEPARATOR -->
<!ENTITY InvisibleTimes "⁢" ><!--INVISIBLE TIMES -->
<!ENTITY langle "⟨" ><!--MATHEMATICAL LEFT ANGLE BRACKET -->
<!ENTITY Laplacetrf "ℒ" ><!--SCRIPT CAPITAL L -->
<!ENTITY lbrace "{" ><!--LEFT CURLY BRACKET -->
<!ENTITY lbrack "[" ><!--LEFT SQUARE BRACKET -->
<!ENTITY LeftAngleBracket "⟨" ><!--MATHEMATICAL LEFT ANGLE BRACKET -->
<!ENTITY leftarrow "←" ><!--LEFTWARDS ARROW -->
<!ENTITY Leftarrow "⇐" ><!--LEFTWARDS DOUBLE ARROW -->
<!ENTITY LeftArrow "←" ><!--LEFTWARDS ARROW -->
<!ENTITY LeftArrowBar "⇤" ><!--LEFTWARDS ARROW TO BAR -->
<!ENTITY LeftArrowRightArrow "⇆" ><!--LEFTWARDS ARROW OVER RIGHTWARDS ARROW -->
<!ENTITY leftarrowtail "↢" ><!--LEFTWARDS ARROW WITH TAIL -->

<!ENTITY LeftCeiling "⌈" ><!--LEFT CEILING -->
<!ENTITY LeftDoubleBracket "⟦" ><!--MATHEMATICAL LEFT WHITE SQUARE BRACKET -->
<!ENTITY LeftDownVector "⇃" ><!--DOWNWARDS HARPOON WITH BARB LEFTWARDS -->
<!ENTITY LeftFloor "⌊" ><!--LEFT FLOOR -->
<!ENTITY leftharpoondown "↽" ><!--LEFTWARDS HARPOON WITH BARB DOWNWARDS -->
<!ENTITY leftharpoonup "↼" ><!--LEFTWARDS HARPOON WITH BARB UPWARDS -->
<!ENTITY leftleftarrows "⇇" ><!--LEFTWARDS PAIRED ARROWS -->
<!ENTITY leftrightarrow "↔" ><!--LEFT RIGHT ARROW -->
<!ENTITY Leftrightarrow "⇔" ><!--LEFT RIGHT DOUBLE ARROW -->
<!ENTITY LeftRightArrow "↔" ><!--LEFT RIGHT ARROW -->
<!ENTITY leftrightarrows "⇆" ><!--LEFTWARDS ARROW OVER RIGHTWARDS ARROW -->
<!ENTITY leftrightharpoons "⇋" ><!--LEFTWARDS HARPOON OVER RIGHTWARDS HARPOON -
->
<!ENTITY leftrightsquigarrow "↭" ><!--LEFT RIGHT WAVE ARROW -->
<!ENTITY LeftTee "⊣" ><!--LEFT TACK -->
<!ENTITY LeftTeeArrow "↤" ><!--LEFTWARDS ARROW FROM BAR -->
<!ENTITY leftthreetimes "⋋" ><!--LEFT SEMIDIRECT PRODUCT -->
<!ENTITY LeftTriangle "⊲" ><!--NORMAL SUBGROUP OF -->
<!ENTITY LeftTriangleEqual "⊴" ><!--NORMAL SUBGROUP OF OR EQUAL TO -->
<!ENTITY LeftUpVector "↿" ><!--UPWARDS HARPOON WITH BARB LEFTWARDS -->
<!ENTITY LeftVector "↼" ><!--LEFTWARDS HARPOON WITH BARB UPWARDS -->
<!ENTITY leq "≤" ><!--LESS-THAN OR EQUAL TO -->
<!ENTITY leqq "≦" ><!--LESS-THAN OVER EQUAL TO -->
<!ENTITY leqslant "⩽" ><!--LESS-THAN OR SLANTED EQUAL TO -->
<!ENTITY lessapprox "⪅" ><!--LESS-THAN OR APPROXIMATE -->
<!ENTITY lessdot "⋖" ><!--LESS-THAN WITH DOT -->
<!ENTITY lesseqgtr "⋚" ><!--LESS-THAN EQUAL TO OR GREATER-THAN -->
<!ENTITY lesseqqgtr "⪋" ><!--LESS-THAN ABOVE DOUBLE-LINE EQUAL ABOVE
GREATER-THAN -->
<!ENTITY LessEqualGreater "⋚" ><!--LESS-THAN EQUAL TO OR GREATER-THAN -->
<!ENTITY LessFullEqual "≦" ><!--LESS-THAN OVER EQUAL TO -->
<!ENTITY LessGreater "≶" ><!--LESS-THAN OR GREATER-THAN -->
<!ENTITY lessgtr "≶" ><!--LESS-THAN OR GREATER-THAN -->
<!ENTITY lesssim "≲" ><!--LESS-THAN OR EQUIVALENT TO -->
<!ENTITY LessSlantEqual "⩽" ><!--LESS-THAN OR SLANTED EQUAL TO -->
<!ENTITY LessTilde "≲" ><!--LESS-THAN OR EQUIVALENT TO -->
<!ENTITY ll "≪" ><!--MUCH LESS-THAN -->
<!ENTITY llcorner "⌞" ><!--BOTTOM LEFT CORNER -->
<!ENTITY Lleftarrow "⇚" ><!--LEFTWARDS TRIPLE ARROW -->
<!ENTITY lmoustache "⎰" ><!--UPPER LEFT OR LOWER RIGHT CURLY BRACKET SECTION
-->
<!ENTITY lnapprox "⪉" ><!--LESS-THAN AND NOT APPROXIMATE -->
<!ENTITY lneq "⪇" ><!--LESS-THAN AND SINGLE-LINE NOT EQUAL TO -->
<!ENTITY lneqq "≨" ><!--LESS-THAN BUT NOT EQUAL TO -->
<!ENTITY longleftarrow "⟵" ><!--LONG LEFTWARDS ARROW -->
<!ENTITY Longleftarrow "⟸" ><!--LONG LEFTWARDS DOUBLE ARROW -->
<!ENTITY LongLeftArrow "⟵" ><!--LONG LEFTWARDS ARROW -->
<!ENTITY longleftrightarrow "⟷" ><!--LONG LEFT RIGHT ARROW -->

<!ENTITY Longleftrightarrow "⟺" ><!--LONG LEFT RIGHT DOUBLE ARROW -->
<!ENTITY LongLeftRightArrow "⟷" ><!--LONG LEFT RIGHT ARROW -->
<!ENTITY longmapsto "⟼" ><!--LONG RIGHTWARDS ARROW FROM BAR -->
<!ENTITY longrightarrow "⟶" ><!--LONG RIGHTWARDS ARROW -->
<!ENTITY Longrightarrow "⟹" ><!--LONG RIGHTWARDS DOUBLE ARROW -->
<!ENTITY LongRightArrow "⟶" ><!--LONG RIGHTWARDS ARROW -->
<!ENTITY looparrowleft "↫" ><!--LEFTWARDS ARROW WITH LOOP -->
<!ENTITY looparrowright "↬" ><!--RIGHTWARDS ARROW WITH LOOP -->
<!ENTITY LowerLeftArrow "↙" ><!--SOUTH WEST ARROW -->
<!ENTITY LowerRightArrow "↘" ><!--SOUTH EAST ARROW -->
<!ENTITY lozenge "◊" ><!--LOZENGE -->
<!ENTITY lrcorner "⌟" ><!--BOTTOM RIGHT CORNER -->
<!ENTITY Lsh "↰" ><!--UPWARDS ARROW WITH TIP LEFTWARDS -->
<!ENTITY lvertneqq "≨︀" ><!--LESS-THAN BUT NOT EQUAL TO - with vertical stroke -->
<!ENTITY maltese "✠" ><!--MALTESE CROSS -->
<!ENTITY mapsto "↦" ><!--RIGHTWARDS ARROW FROM BAR -->
<!ENTITY measuredangle "∡" ><!--MEASURED ANGLE -->
<!ENTITY Mellintrf "ℳ" ><!--SCRIPT CAPITAL M -->
<!ENTITY MinusPlus "∓" ><!--MINUS-OR-PLUS SIGN -->
<!ENTITY mp "∓" ><!--MINUS-OR-PLUS SIGN -->
<!ENTITY multimap "⊸" ><!--MULTIMAP -->
<!ENTITY napprox "≉" ><!--NOT ALMOST EQUAL TO -->
<!ENTITY natural "♮" ><!--MUSIC NATURAL SIGN -->
<!ENTITY naturals "ℕ" ><!--DOUBLE-STRUCK CAPITAL N -->
<!ENTITY nearrow "↗" ><!--NORTH EAST ARROW -->
<!ENTITY NegativeMediumSpace "" ><!--ZERO WIDTH SPACE -->
<!ENTITY NegativeThickSpace "" ><!--ZERO WIDTH SPACE -->
<!ENTITY NegativeThinSpace "" ><!--ZERO WIDTH SPACE -->
<!ENTITY NegativeVeryThinSpace "" ><!--ZERO WIDTH SPACE -->
<!ENTITY NestedGreaterGreater "≫" ><!--MUCH GREATER-THAN -->
<!ENTITY NestedLessLess "≪" ><!--MUCH LESS-THAN -->
<!ENTITY nexists "∄" ><!--THERE DOES NOT EXIST -->
<!ENTITY ngeq "≱" ><!--NEITHER GREATER-THAN NOR EQUAL TO -->
<!ENTITY ngeqq "≧̸" ><!--GREATER-THAN OVER EQUAL TO with slash -->
<!ENTITY ngeqslant "⩾̸" ><!--GREATER-THAN OR SLANTED EQUAL TO with slash -->
<!ENTITY ngtr "≯" ><!--NOT GREATER-THAN -->
<!ENTITY nleftarrow "↚" ><!--LEFTWARDS ARROW WITH STROKE -->
<!ENTITY nLeftarrow "⇍" ><!--LEFTWARDS DOUBLE ARROW WITH STROKE -->
<!ENTITY nleftrightarrow "↮" ><!--LEFT RIGHT ARROW WITH STROKE -->
<!ENTITY nLeftrightarrow "⇎" ><!--LEFT RIGHT DOUBLE ARROW WITH STROKE -->
<!ENTITY nleq "≰" ><!--NEITHER LESS-THAN NOR EQUAL TO -->
<!ENTITY nleqq "≦̸" ><!--LESS-THAN OVER EQUAL TO with slash -->
<!ENTITY nleqslant "⩽̸" ><!--LESS-THAN OR SLANTED EQUAL TO with slash -->
<!ENTITY nless "≮" ><!--NOT LESS-THAN -->
<!ENTITY NonBreakingSpace " " ><!--NO-BREAK SPACE -->
<!ENTITY NotCongruent "≢" ><!--NOT IDENTICAL TO -->

<!ENTITY NotDoubleVerticalBar "∦" ><!--NOT PARALLEL TO -->
<!ENTITY NotElement "∉" ><!--NOT AN ELEMENT OF -->
<!ENTITY NotEqual "≠" ><!--NOT EQUAL TO -->
<!ENTITY NotEqualTilde "≂̸" ><!--MINUS TILDE with slash -->
<!ENTITY NotExists "∄" ><!--THERE DOES NOT EXIST -->
<!ENTITY NotGreater "≯" ><!--NOT GREATER-THAN -->
<!ENTITY NotGreaterEqual "≱" ><!--NEITHER GREATER-THAN NOR EQUAL TO -->
<!ENTITY NotGreaterFullEqual "≦̸" ><!--LESS-THAN OVER EQUAL TO with slash -->
<!ENTITY NotGreaterGreater "≫̸" ><!--MUCH GREATER THAN with slash -->
<!ENTITY NotGreaterLess "≹" ><!--NEITHER GREATER-THAN NOR LESS-THAN -->
<!ENTITY NotGreaterSlantEqual "⩾̸" ><!--GREATER-THAN OR SLANTED EQUAL TO with slash -->
<!ENTITY NotGreaterTilde "≵" ><!--NEITHER GREATER-THAN NOR EQUIVALENT TO -->
<!ENTITY NotHumpDownHump "≎̸" ><!--GEOMETRICALLY EQUIVALENT TO with slash -->
<!ENTITY NotLeftTriangle "⋪" ><!--NOT NORMAL SUBGROUP OF -->
<!ENTITY NotLeftTriangleEqual "⋬" ><!--NOT NORMAL SUBGROUP OF OR EQUAL TO -->
<!ENTITY NotLess "≮" ><!--NOT LESS-THAN -->
<!ENTITY NotLessEqual "≰" ><!--NEITHER LESS-THAN NOR EQUAL TO -->
<!ENTITY NotLessGreater "≸" ><!--NEITHER LESS-THAN NOR GREATER-THAN -->
<!ENTITY NotLessLess "≪̸" ><!--MUCH LESS THAN with slash -->
<!ENTITY NotLessSlantEqual "⩽̸" ><!--LESS-THAN OR SLANTED EQUAL TO with slash -->
<!ENTITY NotLessTilde "≴" ><!--NEITHER LESS-THAN NOR EQUIVALENT TO -->
<!ENTITY NotPrecedes "⊀" ><!--DOES NOT PRECEDE -->
<!ENTITY NotPrecedesEqual "⪯̸" ><!--PRECEDES ABOVE SINGLE-LINE EQUALS SIGN with slash -->
<!ENTITY NotPrecedesSlantEqual "⋠" ><!--DOES NOT PRECEDE OR EQUAL -->
<!ENTITY NotReverseElement "∌" ><!--DOES NOT CONTAIN AS MEMBER -->
<!ENTITY NotRightTriangle "⋫" ><!--DOES NOT CONTAIN AS NORMAL SUBGROUP -->
<!ENTITY NotRightTriangleEqual "⋭" ><!--DOES NOT CONTAIN AS NORMAL SUBGROUP OR EQUAL -->
<!ENTITY NotSquareSubsetEqual "⋢" ><!--NOT SQUARE IMAGE OF OR EQUAL TO -->
<!ENTITY NotSquareSupersetEqual "⋣" ><!--NOT SQUARE ORIGINAL OF OR EQUAL TO -->
<!ENTITY NotSubset "⊂⃒" ><!--SUBSET OF with vertical line -->
<!ENTITY NotSubsetEqual "⊈" ><!--NEITHER A SUBSET OF NOR EQUAL TO -->
<!ENTITY NotSucceeds "⊁" ><!--DOES NOT SUCCEED -->
<!ENTITY NotSucceedsEqual "⪰̸" ><!--SUCCEEDS ABOVE SINGLE-LINE EQUALS SIGN with slash -->
<!ENTITY NotSucceedsSlantEqual "⋡" ><!--DOES NOT SUCCEED OR EQUAL -->
<!ENTITY NotSuperset "⊃⃒" ><!--SUPERSET OF with vertical line -->
<!ENTITY NotSupersetEqual "⊉" ><!--NEITHER A SUPERSET OF NOR EQUAL TO -->
<!ENTITY NotTilde "≁" ><!--NOT TILDE -->
<!ENTITY NotTildeEqual "≄" ><!--NOT ASYMPTOTICALLY EQUAL TO -->
<!ENTITY NotTildeFullEqual "≇" ><!--NEITHER APPROXIMATELY NOR ACTUALLY EQUAL TO -->
<!ENTITY NotTildeTilde "≉" ><!--NOT ALMOST EQUAL TO -->
<!ENTITY NotVerticalBar "∤" ><!--DOES NOT DIVIDE -->

<!ENTITY nparallel "∦" ><!--NOT PARALLEL TO -->
<!ENTITY nprec "⊀" ><!--DOES NOT PRECEDE -->
<!ENTITY npreceq "⪯̸" ><!--PRECEDES ABOVE SINGLE-LINE EQUALS SIGN with slash -->
<!ENTITY nrightarrow "↛" ><!--RIGHTWARDS ARROW WITH STROKE -->
<!ENTITY nRightarrow "⇏" ><!--RIGHTWARDS DOUBLE ARROW WITH STROKE -->
<!ENTITY nshortmid "∤" ><!--DOES NOT DIVIDE -->
<!ENTITY nshortparallel "∦" ><!--NOT PARALLEL TO -->
<!ENTITY nsimeq "≄" ><!--NOT ASYMPTOTICALLY EQUAL TO -->
<!ENTITY nsubset "⊂⃒" ><!--SUBSET OF with vertical line -->
<!ENTITY nsubseteq "⊈" ><!--NEITHER A SUBSET OF NOR EQUAL TO -->
<!ENTITY nsubseteqq "⫅̸" ><!--SUBSET OF ABOVE EQUALS SIGN with slash -->
<!ENTITY nsucc "⊁" ><!--DOES NOT SUCCEED -->
<!ENTITY nsucceq "⪰̸" ><!--SUCCEEDS ABOVE SINGLE-LINE EQUALS SIGN with slash -->
<!ENTITY nsupset "⊃⃒" ><!--SUPERSET OF with vertical line -->
<!ENTITY nsupseteq "⊉" ><!--NEITHER A SUPERSET OF NOR EQUAL TO -->
<!ENTITY nsupseteqq "⫆̸" ><!--SUPERSET OF ABOVE EQUALS SIGN with slash -->
<!ENTITY ntriangleleft "⋪" ><!--NOT NORMAL SUBGROUP OF -->
<!ENTITY ntrianglelefteq "⋬" ><!--NOT NORMAL SUBGROUP OF OR EQUAL TO -->
<!ENTITY ntriangleright "⋫" ><!--DOES NOT CONTAIN AS NORMAL SUBGROUP -->
<!ENTITY ntrianglerighteq "⋭" ><!--DOES NOT CONTAIN AS NORMAL SUBGROUP OR EQUAL -->
<!ENTITY nwarrow "↖" ><!--NORTH WEST ARROW -->
<!ENTITY oint "∮" ><!--CONTOUR INTEGRAL -->
<!ENTITY OpenCurlyDoubleQuote "“" ><!--LEFT DOUBLE QUOTATION MARK -->
<!ENTITY OpenCurlyQuote "‘" ><!--LEFT SINGLE QUOTATION MARK -->
<!ENTITY orderof "ℴ" ><!--SCRIPT SMALL O -->
<!ENTITY parallel "∥" ><!--PARALLEL TO -->
<!ENTITY PartialD "∂" ><!--PARTIAL DIFFERENTIAL -->
<!ENTITY pitchfork "⋔" ><!--PITCHFORK -->
<!ENTITY PlusMinus "±" ><!--PLUS-MINUS SIGN -->
<!ENTITY pm "±" ><!--PLUS-MINUS SIGN -->
<!ENTITY Poincareplane "ℌ" ><!--BLACK-LETTER CAPITAL H -->
<!ENTITY prec "≺" ><!--PRECEDES -->
<!ENTITY precapprox "⪷" ><!--PRECEDES ABOVE ALMOST EQUAL TO -->
<!ENTITY preccurlyeq "≼" ><!--PRECEDES OR EQUAL TO -->
<!ENTITY Precedes "≺" ><!--PRECEDES -->
<!ENTITY PrecedesEqual "⪯" ><!--PRECEDES ABOVE SINGLE-LINE EQUALS SIGN -->
<!ENTITY PrecedesSlantEqual "≼" ><!--PRECEDES OR EQUAL TO -->
<!ENTITY PrecedesTilde "≾" ><!--PRECEDES OR EQUIVALENT TO -->
<!ENTITY preceq "⪯" ><!--PRECEDES ABOVE SINGLE-LINE EQUALS SIGN -->
<!ENTITY precnapprox "⪹" ><!--PRECEDES ABOVE NOT ALMOST EQUAL TO -->
<!ENTITY precneqq "⪵" ><!--PRECEDES ABOVE NOT EQUAL TO -->
<!ENTITY precnsim "⋨" ><!--PRECEDES BUT NOT EQUIVALENT TO -->
<!ENTITY precsim "≾" ><!--PRECEDES OR EQUIVALENT TO -->
<!ENTITY primes "ℙ" ><!--DOUBLE-STRUCK CAPITAL P -->
<!ENTITY Proportion "∷" ><!--PROPORTION -->

<!ENTITY Proportional "∝" ><!--PROPORTIONAL TO -->
<!ENTITY propto "∝" ><!--PROPORTIONAL TO -->
<!ENTITY quaternions "ℍ" ><!--DOUBLE-STRUCK CAPITAL H -->
<!ENTITY questeq "≟" ><!--QUESTIONED EQUAL TO -->
<!ENTITY rangle "⟩" ><!--MATHEMATICAL RIGHT ANGLE BRACKET -->
<!ENTITY rationals "ℚ" ><!--DOUBLE-STRUCK CAPITAL Q -->
<!ENTITY rbrace "}" ><!--RIGHT CURLY BRACKET -->
<!ENTITY rbrack "]" ><!--RIGHT SQUARE BRACKET -->
<!ENTITY Re "ℜ" ><!--BLACK-LETTER CAPITAL R -->
<!ENTITY realine "ℛ" ><!--SCRIPT CAPITAL R -->
<!ENTITY realpart "ℜ" ><!--BLACK-LETTER CAPITAL R -->
<!ENTITY reals "ℝ" ><!--DOUBLE-STRUCK CAPITAL R -->
<!ENTITY ReverseElement "∋" ><!--CONTAINS AS MEMBER -->
<!ENTITY ReverseEquilibrium "⇋" ><!--LEFTWARDS HARPOON OVER RIGHTWARDS HARPOON -->
<!ENTITY ReverseUpEquilibrium "⥯" ><!--DOWNWARDS HARPOON WITH BARB LEFT BESIDE UPWARDS HARPOON WITH BARB RIGHT -->
<!ENTITY RightAngleBracket "⟩" ><!--MATHEMATICAL RIGHT ANGLE BRACKET -->
<!ENTITY rightarrow "→" ><!--RIGHTWARDS ARROW -->
<!ENTITY Rightarrow "⇒" ><!--RIGHTWARDS DOUBLE ARROW -->
<!ENTITY RightArrow "→" ><!--RIGHTWARDS ARROW -->
<!ENTITY RightArrowBar "⇥" ><!--RIGHTWARDS ARROW TO BAR -->
<!ENTITY RightArrowLeftArrow "⇄" ><!--RIGHTWARDS ARROW OVER LEFTWARDS ARROW -->
<!ENTITY rightarrowtail "↣" ><!--RIGHTWARDS ARROW WITH TAIL -->
<!ENTITY RightCeiling "⌉" ><!--RIGHT CEILING -->
<!ENTITY RightDoubleBracket "⟧" ><!--MATHEMATICAL RIGHT WHITE SQUARE BRACKET -->
<!ENTITY RightDownVector "⇂" ><!--DOWNWARDS HARPOON WITH BARB RIGHTWARDS -->
<!ENTITY RightFloor "⌋" ><!--RIGHT FLOOR -->
<!ENTITY rightharpoondown "⇁" ><!--RIGHTWARDS HARPOON WITH BARB DOWNWARDS -->
<!ENTITY rightharpoonup "⇀" ><!--RIGHTWARDS HARPOON WITH BARB UPWARDS -->
<!ENTITY rightleftarrows "⇄" ><!--RIGHTWARDS ARROW OVER LEFTWARDS ARROW -->
<!ENTITY rightleftharpoons "⇌" ><!--RIGHTWARDS HARPOON OVER LEFTWARDS HARPOON -->
<!ENTITY rightrightarrows "⇉" ><!--RIGHTWARDS PAIRED ARROWS -->
<!ENTITY rightsquigarrow "↝" ><!--RIGHTWARDS WAVE ARROW -->
<!ENTITY RightTee "⊢" ><!--RIGHT TACK -->
<!ENTITY RightTeeArrow "↦" ><!--RIGHTWARDS ARROW FROM BAR -->
<!ENTITY rightthreetimes "⋌" ><!--RIGHT SEMIDIRECT PRODUCT -->
<!ENTITY RightTriangle "⊳" ><!--CONTAINS AS NORMAL SUBGROUP -->
<!ENTITY RightTriangleEqual "⊵" ><!--CONTAINS AS NORMAL SUBGROUP OR EQUAL TO -->
<!ENTITY RightUpVector "↾" ><!--UPWARDS HARPOON WITH BARB RIGHTWARDS -->
<!ENTITY RightVector "⇀" ><!--RIGHTWARDS HARPOON WITH BARB UPWARDS -->
<!ENTITY risingdotseq "≓" ><!--IMAGE OF OR APPROXIMATELY EQUAL TO -->
<!ENTITY rmoustache "⎱" ><!--UPPER RIGHT OR LOWER LEFT CURLY BRACKET SECTION -->
<!ENTITY Rrightarrow "⇛" ><!--RIGHTWARDS TRIPLE ARROW -->
<!ENTITY Rsh "↱" ><!--UPWARDS ARROW WITH TIP RIGHTWARDS -->

<!ENTITY searrow "↘" ><!--SOUTH EAST ARROW -->
<!ENTITY setminus "∖" ><!--SET MINUS -->
<!ENTITY ShortDownArrow "↓" ><!--DOWNWARDS ARROW -->
<!ENTITY ShortLeftArrow "←" ><!--LEFTWARDS ARROW -->
<!ENTITY shortmid "∣" ><!--DIVIDES -->
<!ENTITY shortparallel "∥" ><!--PARALLEL TO -->
<!ENTITY ShortRightArrow "→" ><!--RIGHTWARDS ARROW -->
<!ENTITY ShortUpArrow "↑" ><!--UPWARDS ARROW -->
<!ENTITY simeq "≃" ><!--ASYMPTOTICALLY EQUAL TO -->
<!ENTITY SmallCircle "∘" ><!--RING OPERATOR -->
<!ENTITY smallsetminus "∖" ><!--SET MINUS -->
<!ENTITY spadesuit "♠" ><!--BLACK SPADE SUIT -->
<!ENTITY Sqrt "√" ><!--SQUARE ROOT -->
<!ENTITY sqsubset "⊏" ><!--SQUARE IMAGE OF -->
<!ENTITY sqsubseteq "⊑" ><!--SQUARE IMAGE OF OR EQUAL TO -->
<!ENTITY sqsupset "⊐" ><!--SQUARE ORIGINAL OF -->
<!ENTITY sqsupseteq "⊒" ><!--SQUARE ORIGINAL OF OR EQUAL TO -->
<!ENTITY Square "□" ><!--WHITE SQUARE -->
<!ENTITY SquareIntersection "⊓" ><!--SQUARE CAP -->
<!ENTITY SquareSubset "⊏" ><!--SQUARE IMAGE OF -->
<!ENTITY SquareSubsetEqual "⊑" ><!--SQUARE IMAGE OF OR EQUAL TO -->
<!ENTITY SquareSuperset "⊐" ><!--SQUARE ORIGINAL OF -->
<!ENTITY SquareSupersetEqual "⊒" ><!--SQUARE ORIGINAL OF OR EQUAL TO -->
<!ENTITY SquareUnion "⊔" ><!--SQUARE CUP -->
<!ENTITY Star "⋆" ><!--STAR OPERATOR -->
<!ENTITY straightepsilon "ϵ" ><!--GREEK LUNATE EPSILON SYMBOL -->
<!ENTITY straightphi "ϕ" ><!--GREEK PHI SYMBOL -->
<!ENTITY subset "⊂" ><!--SUBSET OF -->
<!ENTITY Subset "⋐" ><!--DOUBLE SUBSET -->
<!ENTITY subseteq "⊆" ><!--SUBSET OF OR EQUAL TO -->
<!ENTITY subseteqq "⫅" ><!--SUBSET OF ABOVE EQUALS SIGN -->
<!ENTITY SubsetEqual "⊆" ><!--SUBSET OF OR EQUAL TO -->
<!ENTITY subsetneq "⊊" ><!--SUBSET OF WITH NOT EQUAL TO -->
<!ENTITY subsetneqq "⫋" ><!--SUBSET OF ABOVE NOT EQUAL TO -->
<!ENTITY succ "≻" ><!--SUCCEEDS -->
<!ENTITY succapprox "⪸" ><!--SUCCEEDS ABOVE ALMOST EQUAL TO -->
<!ENTITY succcurlyeq "≽" ><!--SUCCEEDS OR EQUAL TO -->
<!ENTITY Succeeds "≻" ><!--SUCCEEDS -->
<!ENTITY SucceedsEqual "⪰" ><!--SUCCEEDS ABOVE SINGLE-LINE EQUALS SIGN -->
<!ENTITY SucceedsSlantEqual "≽" ><!--SUCCEEDS OR EQUAL TO -->
<!ENTITY SucceedsTilde "≿" ><!--SUCCEEDS OR EQUIVALENT TO -->
<!ENTITY succeq "⪰" ><!--SUCCEEDS ABOVE SINGLE-LINE EQUALS SIGN -->
<!ENTITY succnapprox "⪺" ><!--SUCCEEDS ABOVE NOT ALMOST EQUAL TO -->
<!ENTITY succneqq "⪶" ><!--SUCCEEDS ABOVE NOT EQUAL TO -->
<!ENTITY succnsim "⋩" ><!--SUCCEEDS BUT NOT EQUIVALENT TO -->
<!ENTITY succsim "≿" ><!--SUCCEEDS OR EQUIVALENT TO -->
<!ENTITY SuchThat "∋" ><!--CONTAINS AS MEMBER -->
<!ENTITY Sum "∑" ><!--N-ARY SUMMATION -->

<!ENTITY Superset "⊃" ><!--SUPERSET OF -->
<!ENTITY SupersetEqual "⊇" ><!--SUPERSET OF OR EQUAL TO -->
<!ENTITY supset "⊃" ><!--SUPERSET OF -->
<!ENTITY Supset "⋑" ><!--DOUBLE SUPERSET -->
<!ENTITY supseteq "⊇" ><!--SUPERSET OF OR EQUAL TO -->
<!ENTITY supseteqq "⫆" ><!--SUPERSET OF ABOVE EQUALS SIGN -->
<!ENTITY supsetneq "⊋" ><!--SUPERSET OF WITH NOT EQUAL TO -->
<!ENTITY supsetneqq "⫌" ><!--SUPERSET OF ABOVE NOT EQUAL TO -->
<!ENTITY swarrow "↙" ><!--SOUTH WEST ARROW -->
<!ENTITY therefore "∴" ><!--THEREFORE -->
<!ENTITY Therefore "∴" ><!--THEREFORE -->
<!ENTITY thickapprox "≈" ><!--ALMOST EQUAL TO -->
<!ENTITY thicksim "∼" ><!--TILDE OPERATOR -->
<!ENTITY ThinSpace " " ><!--THIN SPACE -->
<!ENTITY Tilde "∼" ><!--TILDE OPERATOR -->
<!ENTITY TildeEqual "≃" ><!--ASYMPTOTICALLY EQUAL TO -->
<!ENTITY TildeFullEqual "≅" ><!--APPROXIMATELY EQUAL TO -->
<!ENTITY TildeTilde "≈" ><!--ALMOST EQUAL TO -->
<!ENTITY toea "⤨" ><!--NORTH EAST ARROW AND SOUTH EAST ARROW -->
<!ENTITY tosa "⤩" ><!--SOUTH EAST ARROW AND SOUTH WEST ARROW -->
<!ENTITY triangle "▵" ><!--WHITE UP-POINTING SMALL TRIANGLE -->
<!ENTITY triangledown "▿" ><!--WHITE DOWN-POINTING SMALL TRIANGLE -->
<!ENTITY triangleleft "◃" ><!--WHITE LEFT-POINTING SMALL TRIANGLE -->
<!ENTITY trianglelefteq "⊴" ><!--NORMAL SUBGROUP OF OR EQUAL TO -->
<!ENTITY triangleq "≜" ><!--DELTA EQUAL TO -->
<!ENTITY triangleright "▹" ><!--WHITE RIGHT-POINTING SMALL TRIANGLE -->
<!ENTITY trianglerighteq "⊵" ><!--CONTAINS AS NORMAL SUBGROUP OR EQUAL TO -->
<!ENTITY TripleDot " ⃛" ><!--COMBINING THREE DOTS ABOVE -->
<!ENTITY twoheadleftarrow "↞" ><!--LEFTWARDS TWO HEADED ARROW -->
<!ENTITY twoheadrightarrow "↠" ><!--RIGHTWARDS TWO HEADED ARROW -->
<!ENTITY ulcorner "⌜" ><!--TOP LEFT CORNER -->
<!ENTITY Union "⋃" ><!--N-ARY UNION -->
<!ENTITY UnionPlus "⊎" ><!--MULTISET UNION -->
<!ENTITY uparrow "↑" ><!--UPWARDS ARROW -->
<!ENTITY Uparrow "⇑" ><!--UPWARDS DOUBLE ARROW -->
<!ENTITY UpArrow "↑" ><!--UPWARDS ARROW -->
<!ENTITY UpArrowDownArrow "⇅" ><!--UPWARDS ARROW LEFTWARDS OF DOWNWARDS ARROW -->
<!ENTITY updownarrow "↕" ><!--UP DOWN ARROW -->
<!ENTITY Updownarrow "⇕" ><!--UP DOWN DOUBLE ARROW -->
<!ENTITY UpDownArrow "↕" ><!--UP DOWN ARROW -->
<!ENTITY UpEquilibrium "⥮" ><!--UPWARDS HARPOON WITH BARB LEFT BESIDE DOWNWARDS HARPOON WITH BARB RIGHT -->
<!ENTITY upharpoonleft "↿" ><!--UPWARDS HARPOON WITH BARB LEFTWARDS -->
<!ENTITY upharpoonright "↾" ><!--UPWARDS HARPOON WITH BARB RIGHTWARDS -->
<!ENTITY UpperLeftArrow "↖" ><!--NORTH WEST ARROW -->
<!ENTITY UpperRightArrow "↗" ><!--NORTH EAST ARROW -->
<!ENTITY upsilon "υ" ><!--GREEK SMALL LETTER UPSILON -->

<!ENTITY UpTee "⊥" ><!--UP TACK -->
<!ENTITY UpTeeArrow "↥" ><!--UPWARDS ARROW FROM BAR -->
<!ENTITY upuparrows "⇈" ><!--UPWARDS PAIRED ARROWS -->
<!ENTITY urcorner "⌝" ><!--TOP RIGHT CORNER -->
<!ENTITY varepsilon "ϵ" ><!--GREEK LUNATE EPSILON SYMBOL -->
<!ENTITY varkappa "ϰ" ><!--GREEK KAPPA SYMBOL -->
<!ENTITY varnothing "∅" ><!--EMPTY SET -->
<!ENTITY varphi "ϕ" ><!--GREEK PHI SYMBOL -->
<!ENTITY varpi "ϖ" ><!--GREEK PI SYMBOL -->
<!ENTITY varpropto "∝" ><!--PROPORTIONAL TO -->
<!ENTITY varrho "ϱ" ><!--GREEK RHO SYMBOL -->
<!ENTITY varsigma "ς" ><!--GREEK SMALL LETTER FINAL SIGMA -->
<!ENTITY varsubsetneq "⊊︀" ><!--SUBSET OF WITH NOT EQUAL TO - variant with stroke through bottom members -->
<!ENTITY varsubsetneqq "⫋︀" ><!--SUBSET OF ABOVE NOT EQUAL TO - variant with stroke through bottom members -->
<!ENTITY varsupsetneq "⊋︀" ><!--SUPERSET OF WITH NOT EQUAL TO - variant with stroke through bottom members -->
<!ENTITY varsupsetneqq "⫌︀" ><!--SUPERSET OF ABOVE NOT EQUAL TO - variant with stroke through bottom members -->
<!ENTITY vartheta "ϑ" ><!--GREEK THETA SYMBOL -->
<!ENTITY vartriangleleft "⊲" ><!--NORMAL SUBGROUP OF -->
<!ENTITY vartriangleright "⊳" ><!--CONTAINS AS NORMAL SUBGROUP -->
<!ENTITY vee "∨" ><!--LOGICAL OR -->
<!ENTITY Vee "⋁" ><!--N-ARY LOGICAL OR -->
<!ENTITY vert "|" ><!--VERTICAL LINE -->
<!ENTITY Vert "‖" ><!--DOUBLE VERTICAL LINE -->
<!ENTITY VerticalBar "∣" ><!--DIVIDES -->
<!ENTITY VerticalTilde "≀" ><!--WREATH PRODUCT -->
<!ENTITY VeryThinSpace " " ><!--HAIR SPACE -->
<!ENTITY wedge "∧" ><!--LOGICAL AND -->
<!ENTITY Wedge "⋀" ><!--N-ARY LOGICAL AND -->
<!ENTITY wp "℘" ><!--SCRIPT CAPITAL P -->
<!ENTITY wr "≀" ><!--WREATH PRODUCT -->
<!ENTITY zeetrf "ℨ" ><!--BLACK-LETTER CAPITAL Z -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/mmlalias.ent
No license file was found, but licenses were detected in source scan.

<!--
File isobox.ent produced by the XSL script entities.xsl
from input data in unicode.xml.

Copyright 1998 - 2009 W3C.

Use and distribution of this code are permitted under the terms of the

W3C Software Notice and License.
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

Please report any errors to David Carlisle
via the public W3C list www-math@w3.org.

The numeric character values assigned to each entity
(should) match the Unicode assignments in Unicode 5.x.

Entity names in this file are derived from files carrying the
following notice:

(C) International Organization for Standardization 1986
Permission to copy in any form is granted for use with
conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Box and Line Drawing//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isobox.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isobox PUBLIC
"-//W3C//ENTITIES Box and Line Drawing//EN//XML"
"http://www.w3.org/2003/entities/2007/isobox.ent"
>
%isobox;

-->

<!ENTITY boxdl "┐" ><!--BOX DRAWINGS LIGHT DOWN AND LEFT -->
<!ENTITY boxdL "╕" ><!--BOX DRAWINGS DOWN SINGLE AND LEFT DOUBLE -->
<!ENTITY boxDl "╖" ><!--BOX DRAWINGS DOWN DOUBLE AND LEFT SINGLE -->
<!ENTITY boxDL "╗" ><!--BOX DRAWINGS DOUBLE DOWN AND LEFT -->
<!ENTITY boxdr "┌" ><!--BOX DRAWINGS LIGHT DOWN AND RIGHT -->
<!ENTITY boxdR "╒" ><!--BOX DRAWINGS DOWN SINGLE AND RIGHT DOUBLE -->
<!ENTITY boxDr "╓" ><!--BOX DRAWINGS DOWN DOUBLE AND RIGHT SINGLE -->
<!ENTITY boxDR "╔" ><!--BOX DRAWINGS DOUBLE DOWN AND RIGHT -->
<!ENTITY boxh "─" ><!--BOX DRAWINGS LIGHT HORIZONTAL -->
<!ENTITY boxH "═" ><!--BOX DRAWINGS DOUBLE HORIZONTAL -->

<!ENTITY boxhd "┬" ><!--BOX DRAWINGS LIGHT DOWN AND HORIZONTAL -->
<!ENTITY boxhD "╥" ><!--BOX DRAWINGS DOWN DOUBLE AND HORIZONTAL SINGLE -->
<!ENTITY boxHd "╤" ><!--BOX DRAWINGS DOWN SINGLE AND HORIZONTAL DOUBLE -->
<!ENTITY boxHD "╦" ><!--BOX DRAWINGS DOUBLE DOWN AND HORIZONTAL -->
<!ENTITY boxhu "┴" ><!--BOX DRAWINGS LIGHT UP AND HORIZONTAL -->
<!ENTITY boxhU "╨" ><!--BOX DRAWINGS UP DOUBLE AND HORIZONTAL SINGLE -->
<!ENTITY boxHu "╧" ><!--BOX DRAWINGS UP SINGLE AND HORIZONTAL DOUBLE -->
<!ENTITY boxHU "╩" ><!--BOX DRAWINGS DOUBLE UP AND HORIZONTAL -->
<!ENTITY boxul "┘" ><!--BOX DRAWINGS LIGHT UP AND LEFT -->
<!ENTITY boxuL "╛" ><!--BOX DRAWINGS UP SINGLE AND LEFT DOUBLE -->
<!ENTITY boxUl "╜" ><!--BOX DRAWINGS UP DOUBLE AND LEFT SINGLE -->
<!ENTITY boxUL "╝" ><!--BOX DRAWINGS DOUBLE UP AND LEFT -->
<!ENTITY boxur "└" ><!--BOX DRAWINGS LIGHT UP AND RIGHT -->
<!ENTITY boxuR "╘" ><!--BOX DRAWINGS UP SINGLE AND RIGHT DOUBLE -->
<!ENTITY boxUr "╙" ><!--BOX DRAWINGS UP DOUBLE AND RIGHT SINGLE -->
<!ENTITY boxUR "╚" ><!--BOX DRAWINGS DOUBLE UP AND RIGHT -->
<!ENTITY boxv "│" ><!--BOX DRAWINGS LIGHT VERTICAL -->
<!ENTITY boxV "║" ><!--BOX DRAWINGS DOUBLE VERTICAL -->
<!ENTITY boxvh "┼" ><!--BOX DRAWINGS LIGHT VERTICAL AND HORIZONTAL -->
<!ENTITY boxvH "╪" ><!--BOX DRAWINGS VERTICAL SINGLE AND HORIZONTAL DOUBLE -->
<!ENTITY boxVh "╫" ><!--BOX DRAWINGS VERTICAL DOUBLE AND HORIZONTAL SINGLE -->
<!ENTITY boxVH "╬" ><!--BOX DRAWINGS DOUBLE VERTICAL AND HORIZONTAL -->
<!ENTITY boxvl "┤" ><!--BOX DRAWINGS LIGHT VERTICAL AND LEFT -->
<!ENTITY boxvL "╡" ><!--BOX DRAWINGS VERTICAL SINGLE AND LEFT DOUBLE -->
<!ENTITY boxVl "╢" ><!--BOX DRAWINGS VERTICAL DOUBLE AND LEFT SINGLE -->
<!ENTITY boxVL "╣" ><!--BOX DRAWINGS DOUBLE VERTICAL AND LEFT -->
<!ENTITY boxvr "├" ><!--BOX DRAWINGS LIGHT VERTICAL AND RIGHT -->
<!ENTITY boxvR "╞" ><!--BOX DRAWINGS VERTICAL SINGLE AND RIGHT DOUBLE -->
<!ENTITY boxVr "╟" ><!--BOX DRAWINGS VERTICAL DOUBLE AND RIGHT SINGLE -->
<!ENTITY boxVR "╠" ><!--BOX DRAWINGS DOUBLE VERTICAL AND RIGHT -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isobox.ent
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Inline Style Module ..... -->
<!-- file: xhtml-inlstyle-1.mod

This is XHTML, a reformulation of HTML as a modular XML application.
Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
Revision: $Id: xhtml-inlstyle-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $

```
   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Inline Style 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-inlstyle-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Inline Style

   This module declares the 'style' attribute, used to support inline
   style markup. This module must be instantiated prior to the XHTML
   Common Attributes module in order to be included in %Core.attrib;.
-->

<!ENTITY % style.attrib
   "style       CDATA                #IMPLIED"
>


<!ENTITY % Core.extra.attrib
   "%style.attrib;"
>

<!-- end of xhtml-inlstyle-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-inlstyle-1.mod
No license file was found, but licenses were detected in source scan.

<!--
   * This is a schema for the XML representation of JSON used as the target for the
   * function fn:json-to-xml()
   *
   * The schema is made available under the terms of the W3C software notice and license
   * at http://www.w3.org/Consortium/Legal/copyright-software-19980720
   *
  -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/json.xsd
No license file was found, but licenses were detected in source scan.

dimensional graphics in XML.

   The Scalable Vector Graphics (SVG)
```

Copyright 2001, 2002 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the SVG DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/templates/svg11-template.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Document Metainformation Module ..... -->
<!-- file: xhtml-meta-1.mod

This is XHTML, a reformulation of HTML as a modular XML application.
Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: xhtml-meta-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS XHTML Metainformation 1.0//EN"
SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-meta-1.mod"

Revisions:
(none)
..... -->

<!-- Meta Information

meta

This module declares the meta element type and its attributes,
used to provide declarative document metainformation.
-->

<!-- meta: Generic Metainformation ..... -->

```
<!ENTITY % meta.element  "INCLUDE" >
<![%meta.element;[
<!ENTITY % meta.content  "EMPTY" >
<!ENTITY % meta.qname  "meta" >
<!ELEMENT %meta.qname;  %meta.content; >
<!-- end of meta.element -->]]>

<!ENTITY % meta.attlist  "INCLUDE" >
<![%meta.attlist;[
<!ATTLIST %meta.qname;
    %XHTML.xmlns.attrib;
    %I18n.attrib;
    http-equiv   NMTOKEN               #IMPLIED
    name         NMTOKEN               #IMPLIED
    content      CDATA                 #REQUIRED
    scheme       CDATA                 #IMPLIED
>
<!-- end of meta.attlist -->]]>

<!-- end of xhtml-meta-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-meta-1.mod
No license file was found, but licenses were detected in source scan.

```
//W3C//DTD Specification V2.10//EN"
#       "http://www.w3.org/2002/xmlspec/dtd/2.10/xmlspec.dtd">

PURPOSE:
 This XML DTD is for W3C specifications and other technical reports.
 It is based in part on the TEI Lite and Sweb DTDs.

COPYRIGHT:

 Copyright (C) 2000, 2001, 2002, 2003 Sun Microsystems, Inc. All Rights Reserved.
 This document is governed by the W3C Software License[3] as
 described in the FAQ[4].

   [1] http://www.w3.org/TR/xslt
   [2] http://www.w3.org/XML/1998/06/xmlspec

<!-- ..... -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xmlspec/xmlspec.dtd
No license file was found, but licenses were detected in source scan.

```
<!-- XHTML Character Entities Module  ..... -->
<!-- file: xhtml-charent-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-charent-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Character Entities 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-charent-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Character Entities for XHTML

   This module declares the set of character entities for XHTML,
   including the Latin 1, Symbol and Special character collections.
-->

<!ENTITY % xhtml-lat1
  PUBLIC "-//W3C//ENTITIES Latin 1 for XHTML//EN"
        "xhtml-lat1.ent" >
%xhtml-lat1;

<!ENTITY % xhtml-symbol
  PUBLIC "-//W3C//ENTITIES Symbols for XHTML//EN"
        "xhtml-symbol.ent" >
%xhtml-symbol;

<!ENTITY % xhtml-special
  PUBLIC "-//W3C//ENTITIES Special for XHTML//EN"
        "xhtml-special.ent" >
%xhtml-special;

<!-- end of xhtml-charent-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-charent-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Color Profile Module ..... -->
<!-- file: svg-profile.mod
```

```
   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-profile.mod,v 1.3 2002/11/14 15:11:03 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Color Profile//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-profile.mod"

   ..................................................................... -->

<!-- Color Profile

     color-profile

   This module declares markup to provide support for color profile.
-->

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.color-profile.qname "color-profile" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >

<!-- SVG.ColorProfile.class ..... -->

<!ENTITY % SVG.ColorProfile.extra.class "" >

<!ENTITY % SVG.ColorProfile.class
  "| %SVG.color-profile.qname; %SVG.ColorProfile.extra.class;"
>

<!-- SVG.ColorProfile.attrib ..... -->

<!ENTITY % SVG.ColorProfile.extra.attrib "" >

<!ENTITY % SVG.ColorProfile.attrib
  "color-profile CDATA #IMPLIED
   %SVG.ColorProfile.extra.attrib;"
>

<!-- color-profile: Color Profile Element ..... -->

<!ENTITY % SVG.color-profile.extra.content "" >
```

```
<!ENTITY % SVG.color-profile.element "INCLUDE" >
<![%SVG.color-profile.element;[
<!ENTITY % SVG.color-profile.content
  "( %SVG.Description.class; %SVG.color-profile.extra.content; )*"
>
<!ELEMENT %SVG.color-profile.qname; %SVG.color-profile.content; >
<!-- end of SVG.color-profile.element -->]]>

<!ENTITY % SVG.color-profile.attlist "INCLUDE" >
<![%SVG.color-profile.attlist;[
<!ATTLIST %SVG.color-profile.qname;
  %SVG.Core.attrib;
  %SVG.XLink.attrib;
  local CDATA #IMPLIED
  name CDATA #REQUIRED
  rendering-intent ( auto | perceptual | relative-colorimetric | saturation |
                     absolute-colorimetric ) 'auto'
>
<!-- end of SVG.color-profile.attlist -->]]>

<!-- end of svg-profile.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-profile.mod
No license file was found, but licenses were detected in source scan.

```
<!--
   File isodia.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

   Entity names in this file are derived from files carrying the
   following notice:

   (C) International Organization for Standardization 1986
   Permission to copy in any form is granted for use with
```

conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Diacritical Marks//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isodia.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isodia PUBLIC
"-//W3C//ENTITIES Diacritical Marks//EN//XML"
"http://www.w3.org/2003/entities/2007/isodia.ent"
>
%isodia;

-->

<!ENTITY acute "´" ><!--ACUTE ACCENT -->
<!ENTITY breve "˘" ><!--BREVE -->
<!ENTITY caron "ˇ" ><!--CARON -->
<!ENTITY cedil "¸" ><!--CEDILLA -->
<!ENTITY circ "ˆ" ><!--MODIFIER LETTER CIRCUMFLEX ACCENT -->
<!ENTITY dblac "˝" ><!--DOUBLE ACUTE ACCENT -->
<!ENTITY die "¨" ><!--DIAERESIS -->
<!ENTITY dot "˙" ><!--DOT ABOVE -->
<!ENTITY grave "`" ><!--GRAVE ACCENT -->
<!ENTITY macr "¯" ><!--MACRON -->
<!ENTITY ogon "˛" ><!--OGONEK -->
<!ENTITY ring "˚" ><!--RING ABOVE -->
<!ENTITY tilde "˜" ><!--SMALL TILDE -->
<!ENTITY uml "¨" ><!--DIAERESIS -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isodia.ent
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML 1.1 Document Model Module ..... -->
<!-- file: xhtml11-model-1.mod

```
    This is XHTML 1.1, a reformulation of HTML as a modular XML application.
    Copyright 1998-2008 W3C (MIT, ERCIM, Keio), All Rights Reserved.
    Revision: $Id: xhtml11-model-1.mod,v 1.18 2009/06/24 17:24:55 ahby Exp $ SMI

    This DTD module is identified by the PUBLIC and SYSTEM identifiers:

      PUBLIC "-//W3C//ENTITIES XHTML 1.1 Document Model 1.0//EN"
      SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml11-model-1.mod"

    Revisions:
    (none)
    ..... -->

<!-- XHTML 1.1 Document Model

    This module describes the groupings of elements that make up
    common content models for XHTML elements.

    XHTML has three basic content models:

        %Inline.mix;  character-level elements
        %Block.mix;   block-like elements, eg., paragraphs and lists
        %Flow.mix;    any block or inline elements

    Any parameter entities declared in this module may be used
    to create element content models, but the above three are
    considered 'global' (insofar as that term applies here).

    The reserved word '#PCDATA' (indicating a text string) is now
    included explicitly with each element declaration that is
    declared as mixed content, as XML requires that this token
    occur first in a content model specification.
-->
<!-- Extending the Model

    While in some cases this module may need to be rewritten to
    accommodate changes to the document model, minor extensions
    may be accomplished by redeclaring any of the three *.extra;
    parameter entities to contain extension element types as follows:

        %Misc.extra;    whose parent may be any block or
                        inline element.

        %Inline.extra;  whose parent may be any inline element.

        %Block.extra;   whose parent may be any block element.

    If used, these parameter entities must be an OR-separated
```

```
   list beginning with an OR separator ("|"), eg., "| a | b | c"

   All block and inline *.class parameter entities not part
   of the *struct.class classes begin with "| " to allow for
   exclusion from mixes.
-->

<!-- ..............  Optional Elements in head  ..................... -->

<!ENTITY % HeadOpts.mix
   "( %script.qname; | %style.qname; | %meta.qname;
    | %link.qname; | %object.qname; )*"
>

<!-- ..................  Miscellaneous Elements  ..................... -->

<!-- ins and del are used to denote editing changes
-->
<!ENTITY % Edit.class "| %ins.qname; | %del.qname;" >

<!-- script and noscript are used to contain scripts
   and alternative content
-->
<!ENTITY % Script.class "| %script.qname; | %noscript.qname;" >

<!ENTITY % Misc.extra "" >

<!-- These elements are neither block nor inline, and can
   essentially be used anywhere in the document body.
-->
<!ENTITY % Misc.class
   "%Edit.class;
    %Script.class;
    %Misc.extra;"
>

<!-- ....................  Inline Elements  ....................... -->

<!ENTITY % InlStruct.class "%br.qname; | %span.qname;" >

<!ENTITY % InlPhras.class
   "| %em.qname; | %strong.qname; | %dfn.qname; | %code.qname;
    | %samp.qname; | %kbd.qname; | %var.qname; | %cite.qname;
    | %abbr.qname; | %acronym.qname; | %q.qname;" >

<!ENTITY % InlPres.class
   "| %tt.qname; | %i.qname; | %b.qname; | %big.qname;
    | %small.qname; | %sub.qname; | %sup.qname;" >
```

```
<!ENTITY % I18n.class "| %bdo.qname;" >

<!ENTITY % Anchor.class "| %a.qname;" >

<!ENTITY % InlSpecial.class
   "| %img.qname; | %map.qname;
    | %object.qname;" >

<!ENTITY % InlForm.class
   "| %input.qname; | %select.qname; | %textarea.qname;
    | %label.qname; | %button.qname;" >

<!ENTITY % Inline.extra "" >

<!ENTITY % Ruby.class "| %ruby.qname;" >

<!-- %Inline.class; includes all inline elements,
   used as a component in mixes
-->
<!ENTITY % Inline.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %I18n.class;
    %Anchor.class;
    %InlSpecial.class;
    %InlForm.class;
    %Ruby.class;
    %Inline.extra;"
>

<!-- %InlNoRuby.class; includes all inline elements
   except ruby, used as a component in mixes
-->
<!ENTITY % InlNoRuby.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %I18n.class;
    %Anchor.class;
    %InlSpecial.class;
    %InlForm.class;
    %Inline.extra;"
>

<!-- %NoRuby.content; includes all inlines except ruby
-->
```

```
<!ENTITY % NoRuby.content
   "( #PCDATA
    | %InlNoRuby.class;
    %Misc.class; )*"
>

<!-- %InlNoAnchor.class; includes all non-anchor inlines,
   used as a component in mixes
-->
<!ENTITY % InlNoAnchor.class
   "%InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %I18n.class;
    %InlSpecial.class;
    %InlForm.class;
    %Ruby.class;
    %Inline.extra;"
>

<!-- %InlNoAnchor.mix; includes all non-anchor inlines
-->
<!ENTITY % InlNoAnchor.mix
   "%InlNoAnchor.class;
    %Misc.class;"
>

<!-- %Inline.mix; includes all inline elements, including %Misc.class;
-->
<!ENTITY % Inline.mix
   "%Inline.class;
    %Misc.class;"
>

<!-- .....................  Block Elements  ..................... -->

<!-- In the HTML 4.0 DTD, heading and list elements were included
   in the %block; parameter entity. The %Heading.class; and
   %List.class; parameter entities must now be included explicitly
   on element declarations where desired.
-->

<!ENTITY % Heading.class
   "%h1.qname; | %h2.qname; | %h3.qname;
    | %h4.qname; | %h5.qname; | %h6.qname;" >

<!ENTITY % List.class "%ul.qname; | %ol.qname; | %dl.qname;" >
```

```
<!ENTITY % Table.class "| %table.qname;" >

<!ENTITY % Form.class  "| %form.qname;" >

<!ENTITY % Fieldset.class  "| %fieldset.qname;" >

<!ENTITY % BlkStruct.class "%p.qname; | %div.qname;" >

<!ENTITY % BlkPhras.class
   "| %pre.qname; | %blockquote.qname; | %address.qname;" >

<!ENTITY % BlkPres.class "| %hr.qname;" >

<!ENTITY % BlkSpecial.class
   "%Table.class;
    %Form.class;
    %Fieldset.class;"
>

<!ENTITY % Block.extra "" >

<!-- %Block.class; includes all block elements,
   used as an component in mixes
-->
<!ENTITY % Block.class
   "%BlkStruct.class;
    %BlkPhras.class;
    %BlkPres.class;
    %BlkSpecial.class;
    %Block.extra;"
>

<!-- %Block.mix; includes all block elements plus %Misc.class;
-->
<!ENTITY % Block.mix
   "%Heading.class;
    | %List.class;
    | %Block.class;
    %Misc.class;"
>

<!-- .....  All Content Elements  ..... -->

<!-- %Flow.mix; includes all text content, block and inline
-->
<!ENTITY % Flow.mix
   "%Heading.class;
    | %List.class;
```

```
    | %Block.class;
    | %Inline.class;
    %Misc.class;"
>

<!-- end of xhtml11-model-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml11-model-1.mod
No license file was found, but licenses were detected in source scan.

```
2003 W3C&#xae; (MIT, ERCIM, Keio), All Rights
   Reserved. W3C liability, trademark, document use and software
   licensing rules apply.

   Permission to use, copy, modify and distribute the MathML 2.0 DTD and
   its accompanying documentation for any purpose and without fee is
   hereby granted in perpetuity, provided that the above copyright notice
   and this paragraph appear in all copies.  The copyright holders make
   no representation about the suitability of the DTD for any purpose.

   It is provided "as is" without expressed or implied warranty.

     Revision:   $Id: mathml2.dtd,v 1.12 2003/11/04 13:14:35 davidc Exp $

   This entity may be identified by the PUBLIC and SYSTEM identifiers:

    PUBLIC "
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml2/mathml2.dtd
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- SVG 1.1 Clip Module ..................................................... -->
<!-- file: svg-clip.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-clip.mod,v 1.3 2002/10/24 17:40:15 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Clip//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-clip.mod"
```

```
    ..... -->

<!-- Clip

      clipPath

   This module declares markup to provide support for clipping.
-->

<!-- 'clip-path' property/attribute value (e.g., 'none', <uri>) -->
<!ENTITY % ClipPathValue.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.clipPath.qname "clipPath" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Clip.class ..... -->

<!ENTITY % SVG.Clip.extra.class "" >

<!ENTITY % SVG.Clip.class
  "| %SVG.clipPath.qname; %SVG.Clip.extra.class;"
>

<!-- SVG.Clip.attrib ..... -->

<!ENTITY % SVG.Clip.extra.attrib "" >

<!ENTITY % SVG.Clip.attrib
  "clip-path %ClipPathValue.datatype; #IMPLIED
   clip-rule %ClipFillRule.datatype; #IMPLIED
```

```
   %SVG.Clip.extra.attrib;"
>

<!-- clipPath: Clip Path Element ..... -->

<!ENTITY % SVG.clipPath.extra.content "" >

<!ENTITY % SVG.clipPath.element "INCLUDE" >
<![%SVG.clipPath.element;[
<!ENTITY % SVG.clipPath.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class; %SVG.Use.class;
     %SVG.Shape.class; %SVG.Text.class; %SVG.clipPath.extra.content; )*)"
>
<!ELEMENT %SVG.clipPath.qname; %SVG.clipPath.content; >
<!-- end of SVG.clipPath.element -->]]>

<!ENTITY % SVG.clipPath.attlist "INCLUDE" >
<![%SVG.clipPath.attlist;[
<!ATTLIST %SVG.clipPath.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Text.attrib;
  %SVG.TextContent.attrib;
  %SVG.Font.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  transform %TransformList.datatype; #IMPLIED
  clipPathUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
>
<!-- end of SVG.clipPath.attlist -->]]>

<!-- end of svg-clip.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-clip.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Link Element Module  ..... -->
```

```
<!-- file: xhtml-link-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-link-1.mod,v 4.1 2001/04/05 06:57:40 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Link Element 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-link-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Link element

      link

   This module declares the link element type and its attributes,
   which could (in principle) be used to define document-level links
   to external resources such as:

   a) for document specific toolbars/menus, e.g. start, contents,
      previous, next, index, end, help
   b) to link to a separate style sheet (rel="stylesheet")
   c) to make a link to a script (rel="script")
   d) by style sheets to control how collections of html nodes are
      rendered into printed documents
   e) to make a link to a printable version of this document
      e.g. a postscript or pdf version (rel="alternate" media="print")
-->

<!-- link: Media-Independent Link ..... -->

<!ENTITY % link.element  "INCLUDE" >
<![%link.element;[
<!ENTITY % link.content  "EMPTY" >
<!ENTITY % link.qname  "link" >
<!ELEMENT %link.qname;  %link.content; >
<!-- end of link.element -->]]>

<!ENTITY % link.attlist  "INCLUDE" >
<![%link.attlist;[
<!ATTLIST %link.qname;
    %Common.attrib;
    charset      %Charset.datatype;       #IMPLIED
    href         %URI.datatype;           #IMPLIED
```

```
    hreflang     %LanguageCode.datatype;  #IMPLIED
    type         %ContentType.datatype;   #IMPLIED
    rel          %LinkTypes.datatype;     #IMPLIED
    rev          %LinkTypes.datatype;     #IMPLIED
    media        %MediaDesc.datatype;     #IMPLIED
>
<!-- end of link.attlist -->]]>

<!-- end of xhtml-link-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-link-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- SVG 1.1 Animation Module ............................................... -->
<!-- file: svg-animation.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-animation.mod,v 1.3 2002/10/24 17:40:14 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Animation//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-animation.mod"

   ....................................................................... -->

<!-- Animation

     animate, set, animateMotion, animateColor, animateTransform, mpath

   This module declares markup to provide support for animation.
-->

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.animate.qname "animate" >
<!ENTITY % SVG.set.qname "set" >
<!ENTITY % SVG.animateMotion.qname "animateMotion" >
<!ENTITY % SVG.animateColor.qname "animateColor" >
<!ENTITY % SVG.animateTransform.qname "animateTransform" >
<!ENTITY % SVG.mpath.qname "mpath" >

<!-- Attribute Collections (Default) ..... -->
```

```
<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.AnimationEvents.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.XLinkRequired.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Animation.class ..... -->

<!ENTITY % SVG.Animation.extra.class "" >

<!ENTITY % SVG.Animation.class
  "%SVG.animate.qname; | %SVG.set.qname; | %SVG.animateMotion.qname; |
   %SVG.animateColor.qname; | %SVG.animateTransform.qname;
   %SVG.Animation.extra.class;"
>

<!-- SVG.Animation.attrib ..... -->

<!ENTITY % SVG.Animation.extra.attrib "" >

<!ENTITY % SVG.Animation.attrib
  "%SVG.XLink.attrib;
   %SVG.Animation.extra.attrib;"
>

<!-- SVG.AnimationAttribute.attrib ..... -->

<!ENTITY % SVG.AnimationAttribute.extra.attrib "" >

<!ENTITY % SVG.AnimationAttribute.attrib
  "attributeName  CDATA  #REQUIRED
   attributeType  CDATA  #IMPLIED
   %SVG.AnimationAttribute.extra.attrib;"
>

<!-- SVG.AnimationTiming.attrib ..... -->

<!ENTITY % SVG.AnimationTiming.extra.attrib "" >

<!ENTITY % SVG.AnimationTiming.attrib
  "begin CDATA #IMPLIED
   dur CDATA #IMPLIED
   end CDATA #IMPLIED
   min CDATA #IMPLIED
   max CDATA #IMPLIED
   restart ( always | never | whenNotActive ) 'always'
   repeatCount CDATA #IMPLIED
```

```
    repeatDur CDATA #IMPLIED
    fill ( remove | freeze ) 'remove'
    %SVG.AnimationTiming.extra.attrib;"
>

<!-- SVG.AnimationValue.attrib ..... -->

<!ENTITY % SVG.AnimationValue.extra.attrib "" >

<!ENTITY % SVG.AnimationValue.attrib
   "calcMode ( discrete | linear | paced | spline ) 'linear'
    values CDATA #IMPLIED
    keyTimes CDATA #IMPLIED
    keySplines CDATA #IMPLIED
    from CDATA #IMPLIED
    to CDATA #IMPLIED
    by CDATA #IMPLIED
    %SVG.AnimationValue.extra.attrib;"
>

<!-- SVG.AnimationAddtion.attrib ..... -->

<!ENTITY % SVG.AnimationAddtion.extra.attrib "" >

<!ENTITY % SVG.AnimationAddtion.attrib
   "additive ( replace | sum ) 'replace'
    accumulate ( none | sum ) 'none'
    %SVG.AnimationAddtion.extra.attrib;"
>

<!-- animate: Animate Element ..... -->

<!ENTITY % SVG.animate.extra.content "" >

<!ENTITY % SVG.animate.element "INCLUDE" >
<![%SVG.animate.element;[
<!ENTITY % SVG.animate.content
   "( %SVG.Description.class; %SVG.animate.extra.content; )*"
>
<!ELEMENT %SVG.animate.qname; %SVG.animate.content; >
<!-- end of SVG.animate.element -->]]>

<!ENTITY % SVG.animate.attlist "INCLUDE" >
<![%SVG.animate.attlist;[
<!ATTLIST %SVG.animate.qname;
   %SVG.Core.attrib;
   %SVG.Conditional.attrib;
   %SVG.AnimationEvents.attrib;
```

```
  %SVG.External.attrib;
  %SVG.Animation.attrib;
  %SVG.AnimationAttribute.attrib;
  %SVG.AnimationTiming.attrib;
  %SVG.AnimationValue.attrib;
  %SVG.AnimationAddtion.attrib;
>
<!-- end of SVG.animate.attlist -->]]>

<!-- set: Set Element ..... -->

<!ENTITY % SVG.set.extra.content "" >

<!ENTITY % SVG.set.element "INCLUDE" >
<![%SVG.set.element;[
<!ENTITY % SVG.set.content
  "( %SVG.Description.class; %SVG.set.extra.content; )*"
>
<!ELEMENT %SVG.set.qname; %SVG.set.content; >
<!-- end of SVG.set.element -->]]>

<!ENTITY % SVG.set.attlist "INCLUDE" >
<![%SVG.set.attlist;[
<!ATTLIST %SVG.set.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.AnimationEvents.attrib;
  %SVG.External.attrib;
  %SVG.Animation.attrib;
  %SVG.AnimationAttribute.attrib;
  %SVG.AnimationTiming.attrib;
  to CDATA #IMPLIED
>
<!-- end of SVG.set.attlist -->]]>

<!-- animateMotion: Animate Motion Element ..... -->

<!ENTITY % SVG.animateMotion.extra.content "" >

<!ENTITY % SVG.animateMotion.element "INCLUDE" >
<![%SVG.animateMotion.element;[
<!ENTITY % SVG.animateMotion.content
  "(( %SVG.Description.class; )*, %SVG.mpath.qname;?
     %SVG.animateMotion.extra.content; )"
>
<!ELEMENT %SVG.animateMotion.qname; %SVG.animateMotion.content; >
<!-- end of SVG.animateMotion.element -->]]>
```

```
<!ENTITY % SVG.animateMotion.attlist "INCLUDE" >
<![%SVG.animateMotion.attlist;[
<!ATTLIST %SVG.animateMotion.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.AnimationEvents.attrib;
  %SVG.External.attrib;
  %SVG.Animation.attrib;
  %SVG.AnimationTiming.attrib;
  %SVG.AnimationAddtion.attrib;
  calcMode ( discrete | linear | paced | spline ) 'paced'
  values CDATA #IMPLIED
  keyTimes CDATA #IMPLIED
  keySplines CDATA #IMPLIED
  from CDATA #IMPLIED
  to CDATA #IMPLIED
  by CDATA #IMPLIED
  path CDATA #IMPLIED
  keyPoints CDATA #IMPLIED
  rotate CDATA #IMPLIED
  origin CDATA #IMPLIED
>
<!-- end of SVG.animateMotion.attlist -->]]>

<!-- animateColor: Animate Color Element ..... -->

<!ENTITY % SVG.animateColor.extra.content "" >

<!ENTITY % SVG.animateColor.element "INCLUDE" >
<![%SVG.animateColor.element;[
<!ENTITY % SVG.animateColor.content
  "( %SVG.Description.class; %SVG.animateColor.extra.content; )*"
>
<!ELEMENT %SVG.animateColor.qname; %SVG.animateColor.content; >
<!-- end of SVG.animateColor.element -->]]>

<!ENTITY % SVG.animateColor.attlist "INCLUDE" >
<![%SVG.animateColor.attlist;[
<!ATTLIST %SVG.animateColor.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.AnimationEvents.attrib;
  %SVG.External.attrib;
  %SVG.Animation.attrib;
  %SVG.AnimationAttribute.attrib;
  %SVG.AnimationTiming.attrib;
  %SVG.AnimationValue.attrib;
  %SVG.AnimationAddtion.attrib;
```

```
>
<!-- end of SVG.animateColor.attlist -->]]>

<!-- animateTransform: Animate Transform Element ..... -->

<!ENTITY % SVG.animateTransform.extra.content "" >

<!ENTITY % SVG.animateTransform.element "INCLUDE" >
<![%SVG.animateTransform.element;[
<!ENTITY % SVG.animateTransform.content
  "( %SVG.Description.class; %SVG.animateTransform.extra.content; )*"
>
<!ELEMENT %SVG.animateTransform.qname; %SVG.animateTransform.content; >
<!-- end of SVG.animateTransform.element -->]]>

<!ENTITY % SVG.animateTransform.attlist "INCLUDE" >
<![%SVG.animateTransform.attlist;[
<!ATTLIST %SVG.animateTransform.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.AnimationEvents.attrib;
  %SVG.External.attrib;
  %SVG.Animation.attrib;
  %SVG.AnimationAttribute.attrib;
  %SVG.AnimationTiming.attrib;
  %SVG.AnimationValue.attrib;
  %SVG.AnimationAddtion.attrib;
  type ( translate | scale | rotate | skewX | skewY ) 'translate'
>
<!-- end of SVG.animateTransform.attlist -->]]>

<!-- mpath: Motion Path Element ..... -->

<!ENTITY % SVG.mpath.extra.content "" >

<!ENTITY % SVG.mpath.element "INCLUDE" >
<![%SVG.mpath.element;[
<!ENTITY % SVG.mpath.content
  "( %SVG.Description.class; %SVG.mpath.extra.content; )*"
>
<!ELEMENT %SVG.mpath.qname; %SVG.mpath.content; >
<!-- end of SVG.mpath.element -->]]>

<!ENTITY % SVG.mpath.attlist "INCLUDE" >
<![%SVG.mpath.attlist;[
<!ATTLIST %SVG.mpath.qname;
  %SVG.Core.attrib;
  %SVG.XLinkRequired.attrib;
```

```
  %SVG.External.attrib;
>
<!-- end of SVG.mpath.attlist -->]]>

<!-- end of svg-animation.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-animation.mod
No license file was found, but licenses were detected in source scan.

```
<!--
   File isogrk3.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

   Entity names in this file are derived from files carrying the
   following notice:

   (C) International Organization for Standardization 1991
   Permission to copy in any form is granted for use with
   conforming SGML systems and applications as defined in
   ISO 8879, provided this notice is included in all copies.

-->


<!--

     Public identifier: -//W3C//ENTITIES Greek Symbols//EN//XML
     System identifier: http://www.w3.org/2003/entities/2007/isogrk3.ent

   The public identifier should always be used verbatim.
   The system identifier may be changed to suit local requirements.

   Typical invocation:
```

```
    <!ENTITY % isogrk3 PUBLIC
      "-//W3C//ENTITIES Greek Symbols//EN//XML"
      "http://www.w3.org/2003/entities/2007/isogrk3.ent"
    >
    %isogrk3;

-->

<!ENTITY alpha      "&#x003B1;" ><!--GREEK SMALL LETTER ALPHA -->
<!ENTITY beta       "&#x003B2;" ><!--GREEK SMALL LETTER BETA -->
<!ENTITY chi        "&#x003C7;" ><!--GREEK SMALL LETTER CHI -->
<!ENTITY delta      "&#x003B4;" ><!--GREEK SMALL LETTER DELTA -->
<!ENTITY Delta      "&#x00394;" ><!--GREEK CAPITAL LETTER DELTA -->
<!ENTITY epsi       "&#x003B5;" ><!--GREEK SMALL LETTER EPSILON -->
<!ENTITY epsiv      "&#x003F5;" ><!--GREEK LUNATE EPSILON SYMBOL -->
<!ENTITY eta        "&#x003B7;" ><!--GREEK SMALL LETTER ETA -->
<!ENTITY gamma      "&#x003B3;" ><!--GREEK SMALL LETTER GAMMA -->
<!ENTITY Gamma      "&#x00393;" ><!--GREEK CAPITAL LETTER GAMMA -->
<!ENTITY gammad     "&#x003DD;" ><!--GREEK SMALL LETTER DIGAMMA -->
<!ENTITY Gammad     "&#x003DC;" ><!--GREEK LETTER DIGAMMA -->
<!ENTITY iota       "&#x003B9;" ><!--GREEK SMALL LETTER IOTA -->
<!ENTITY kappa      "&#x003BA;" ><!--GREEK SMALL LETTER KAPPA -->
<!ENTITY kappav     "&#x003F0;" ><!--GREEK KAPPA SYMBOL -->
<!ENTITY lambda     "&#x003BB;" ><!--GREEK SMALL LETTER LAMDA -->
<!ENTITY Lambda     "&#x0039B;" ><!--GREEK CAPITAL LETTER LAMDA -->
<!ENTITY mu         "&#x003BC;" ><!--GREEK SMALL LETTER MU -->
<!ENTITY nu         "&#x003BD;" ><!--GREEK SMALL LETTER NU -->
<!ENTITY omega      "&#x003C9;" ><!--GREEK SMALL LETTER OMEGA -->
<!ENTITY Omega      "&#x003A9;" ><!--GREEK CAPITAL LETTER OMEGA -->
<!ENTITY phi        "&#x003C6;" ><!--GREEK SMALL LETTER PHI -->
<!ENTITY Phi        "&#x003A6;" ><!--GREEK CAPITAL LETTER PHI -->
<!ENTITY phiv       "&#x003D5;" ><!--GREEK PHI SYMBOL -->
<!ENTITY pi         "&#x003C0;" ><!--GREEK SMALL LETTER PI -->
<!ENTITY Pi         "&#x003A0;" ><!--GREEK CAPITAL LETTER PI -->
<!ENTITY piv        "&#x003D6;" ><!--GREEK PI SYMBOL -->
<!ENTITY psi        "&#x003C8;" ><!--GREEK SMALL LETTER PSI -->
<!ENTITY Psi        "&#x003A8;" ><!--GREEK CAPITAL LETTER PSI -->
<!ENTITY rho        "&#x003C1;" ><!--GREEK SMALL LETTER RHO -->
<!ENTITY rhov       "&#x003F1;" ><!--GREEK RHO SYMBOL -->
<!ENTITY sigma      "&#x003C3;" ><!--GREEK SMALL LETTER SIGMA -->
<!ENTITY Sigma      "&#x003A3;" ><!--GREEK CAPITAL LETTER SIGMA -->
<!ENTITY sigmav     "&#x003C2;" ><!--GREEK SMALL LETTER FINAL SIGMA -->
<!ENTITY tau        "&#x003C4;" ><!--GREEK SMALL LETTER TAU -->
<!ENTITY theta      "&#x003B8;" ><!--GREEK SMALL LETTER THETA -->
<!ENTITY Theta      "&#x00398;" ><!--GREEK CAPITAL LETTER THETA -->
<!ENTITY thetav     "&#x003D1;" ><!--GREEK THETA SYMBOL -->
<!ENTITY upsi       "&#x003C5;" ><!--GREEK SMALL LETTER UPSILON -->
<!ENTITY Upsi       "&#x003D2;" ><!--GREEK UPSILON WITH HOOK SYMBOL -->
```

```
<!ENTITY xi             "&#x003BE;" ><!--GREEK SMALL LETTER XI -->
<!ENTITY Xi             "&#x0039E;" ><!--GREEK CAPITAL LETTER XI -->
<!ENTITY zeta           "&#x003B6;" ><!--GREEK SMALL LETTER ZETA -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isogrk3.ent
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Font Module ................................................. -->
<!-- file: svg-font.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-font.mod,v 1.4 2002/11/14 15:11:02 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Font//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-font.mod"

   ..................................................................... -->

<!-- Font

     font, font-face, glyph, missing-glyph, hkern, vkern, font-face-src,
     font-face-uri, font-face-format, font-face-name, definition-src

   This module declares markup to provide support for template.
-->

<!-- Qualified Names (Default) ....................... -->

<!ENTITY % SVG.font.qname "font" >
<!ENTITY % SVG.font-face.qname "font-face" >
<!ENTITY % SVG.glyph.qname "glyph" >
<!ENTITY % SVG.missing-glyph.qname "missing-glyph" >
<!ENTITY % SVG.hkern.qname "hkern" >
<!ENTITY % SVG.vkern.qname "vkern" >
<!ENTITY % SVG.font-face-src.qname "font-face-src" >
<!ENTITY % SVG.font-face-uri.qname "font-face-uri" >
<!ENTITY % SVG.font-face-format.qname "font-face-format" >
<!ENTITY % SVG.font-face-name.qname "font-face-name" >
<!ENTITY % SVG.definition-src.qname "definition-src" >

<!-- Attribute Collections (Default) ..................... -->
```

```
<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLinkRequired.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Font.class ..................................... -->

<!ENTITY % SVG.Font.extra.class "" >

<!ENTITY % SVG.Font.class
  "| %SVG.font.qname; | %SVG.font-face.qname; %SVG.Font.extra.class;"
>

<!-- SVG.Presentation.attrib ........................... -->

<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
   %SVG.Viewport.attrib;
   %SVG.Text.attrib;
   %SVG.TextContent.attrib;
   %SVG.Font.attrib;
   %SVG.Paint.attrib;
   %SVG.Color.attrib;
   %SVG.Opacity.attrib;
   %SVG.Graphics.attrib;
   %SVG.Marker.attrib;
   %SVG.ColorProfile.attrib;
   %SVG.Gradient.attrib;
   %SVG.Clip.attrib;
```

```
    %SVG.Mask.attrib;
    %SVG.Filter.attrib;
    %SVG.FilterColor.attrib;
    %SVG.Cursor.attrib;
    flood-color %SVGColor.datatype; #IMPLIED
    flood-opacity %OpacityValue.datatype; #IMPLIED
    lighting-color %SVGColor.datatype; #IMPLIED
    %SVG.Presentation.extra.attrib;"
>

<!-- font: Font Element ..... -->

<!ENTITY % SVG.font.extra.content "" >

<!ENTITY % SVG.font.element "INCLUDE" >
<![%SVG.font.element;[
<!ENTITY % SVG.font.content
   "(( %SVG.Description.class; )*, %SVG.font-face.qname;,
      %SVG.missing-glyph.qname;, ( %SVG.glyph.qname; | %SVG.hkern.qname;
     | %SVG.vkern.qname; %SVG.font.extra.content; )*)"
>
<!ELEMENT %SVG.font.qname; %SVG.font.content; >
<!-- end of SVG.font.element -->]]>

<!ENTITY % SVG.font.attlist "INCLUDE" >
<![%SVG.font.attlist;[
<!ATTLIST %SVG.font.qname;
   %SVG.Core.attrib;
   %SVG.Style.attrib;
   %SVG.Presentation.attrib;
   %SVG.External.attrib;
   horiz-origin-x %Number.datatype; #IMPLIED
   horiz-origin-y %Number.datatype; #IMPLIED
   horiz-adv-x %Number.datatype; #REQUIRED
   vert-origin-x %Number.datatype; #IMPLIED
   vert-origin-y %Number.datatype; #IMPLIED
   vert-adv-y %Number.datatype; #IMPLIED
>
<!-- end of SVG.font.attlist -->]]>

<!-- font-face: Font Face Element ..... -->

<!ENTITY % SVG.font-face.extra.content "" >

<!ENTITY % SVG.font-face.element "INCLUDE" >
<![%SVG.font-face.element;[
<!ENTITY % SVG.font-face.content
   "(( %SVG.Description.class; )*, %SVG.font-face-src.qname;?,
```

```
      %SVG.definition-src.qname;? %SVG.font-face.extra.content; )"
>
<!ELEMENT %SVG.font-face.qname; %SVG.font-face.content; >
<!-- end of SVG.font-face.element -->]]>

<!ENTITY % SVG.font-face.attlist "INCLUDE" >
<![%SVG.font-face.attlist;[
<!ATTLIST %SVG.font-face.qname;
  %SVG.Core.attrib;
  font-family CDATA #IMPLIED
  font-style CDATA #IMPLIED
  font-variant CDATA #IMPLIED
  font-weight CDATA #IMPLIED
  font-stretch CDATA #IMPLIED
  font-size CDATA #IMPLIED
  unicode-range CDATA #IMPLIED
  units-per-em %Number.datatype; #IMPLIED
  panose-1 CDATA #IMPLIED
  stemv %Number.datatype; #IMPLIED
  stemh %Number.datatype; #IMPLIED
  slope %Number.datatype; #IMPLIED
  cap-height %Number.datatype; #IMPLIED
  x-height %Number.datatype; #IMPLIED
  accent-height %Number.datatype; #IMPLIED
  ascent %Number.datatype; #IMPLIED
  descent %Number.datatype; #IMPLIED
  widths CDATA #IMPLIED
  bbox CDATA #IMPLIED
  ideographic %Number.datatype; #IMPLIED
  alphabetic %Number.datatype; #IMPLIED
  mathematical %Number.datatype; #IMPLIED
  hanging %Number.datatype; #IMPLIED
  v-ideographic %Number.datatype; #IMPLIED
  v-alphabetic %Number.datatype; #IMPLIED
  v-mathematical %Number.datatype; #IMPLIED
  v-hanging %Number.datatype; #IMPLIED
  underline-position %Number.datatype; #IMPLIED
  underline-thickness %Number.datatype; #IMPLIED
  strikethrough-position %Number.datatype; #IMPLIED
  strikethrough-thickness %Number.datatype; #IMPLIED
  overline-position %Number.datatype; #IMPLIED
  overline-thickness %Number.datatype; #IMPLIED
>
<!-- end of SVG.font-face.attlist -->]]>

<!-- glyph: Glyph Element ..... -->

<!ENTITY % SVG.glyph.extra.content "" >
```

```
<!ENTITY % SVG.glyph.element "INCLUDE" >
<![%SVG.glyph.element;[
<!ENTITY % SVG.glyph.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
    %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
    %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
    %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
    %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
    %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
    %SVG.Font.class; %SVG.glyph.extra.content; )*"
>
<!ELEMENT %SVG.glyph.qname; %SVG.glyph.content; >
<!-- end of SVG.glyph.element -->]]>

<!ENTITY % SVG.glyph.attlist "INCLUDE" >
<![%SVG.glyph.attlist;[
<!ATTLIST %SVG.glyph.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  unicode CDATA #IMPLIED
  glyph-name CDATA #IMPLIED
  d %PathData.datatype; #IMPLIED
  orientation CDATA #IMPLIED
  arabic-form CDATA #IMPLIED
  lang %LanguageCodes.datatype; #IMPLIED
  horiz-adv-x %Number.datatype; #IMPLIED
  vert-origin-x %Number.datatype; #IMPLIED
  vert-origin-y %Number.datatype; #IMPLIED
  vert-adv-y %Number.datatype; #IMPLIED
>
<!-- end of SVG.glyph.attlist -->]]>

<!-- missing-glyph: Missing Glyph Element ..... -->

<!ENTITY % SVG.missing-glyph.extra.content "" >

<!ENTITY % SVG.missing-glyph.element "INCLUDE" >
<![%SVG.missing-glyph.element;[
<!ENTITY % SVG.missing-glyph.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
    %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
    %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
    %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
    %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
    %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
    %SVG.Font.class; %SVG.missing-glyph.extra.content; )*"
```

```
>
<!ELEMENT %SVG.missing-glyph.qname; %SVG.missing-glyph.content; >
<!-- end of SVG.missing-glyph.element -->]]>

<!ENTITY % SVG.missing-glyph.attlist "INCLUDE" >
<![%SVG.missing-glyph.attlist;[
<!ATTLIST %SVG.missing-glyph.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  d %PathData.datatype; #IMPLIED
  horiz-adv-x %Number.datatype; #IMPLIED
  vert-origin-x %Number.datatype; #IMPLIED
  vert-origin-y %Number.datatype; #IMPLIED
  vert-adv-y %Number.datatype; #IMPLIED
>
<!-- end of SVG.missing-glyph.attlist -->]]>

<!-- hkern: Horizontal Kerning Element ..... -->

<!ENTITY % SVG.hkern.element "INCLUDE" >
<![%SVG.hkern.element;[
<!ENTITY % SVG.hkern.content "EMPTY" >
<!ELEMENT %SVG.hkern.qname; %SVG.hkern.content; >
<!-- end of SVG.hkern.element -->]]>

<!ENTITY % SVG.hkern.attlist "INCLUDE" >
<![%SVG.hkern.attlist;[
<!ATTLIST %SVG.hkern.qname;
  %SVG.Core.attrib;
  u1 CDATA #IMPLIED
  g1 CDATA #IMPLIED
  u2 CDATA #IMPLIED
  g2 CDATA #IMPLIED
  k %Number.datatype; #REQUIRED
>
<!-- end of SVG.hkern.attlist -->]]>

<!-- vkern: Vertical Kerning Element ..... -->

<!ENTITY % SVG.vkern.element "INCLUDE" >
<![%SVG.vkern.element;[
<!ENTITY % SVG.vkern.content "EMPTY" >
<!ELEMENT %SVG.vkern.qname; %SVG.vkern.content; >
<!-- end of SVG.vkern.element -->]]>

<!ENTITY % SVG.vkern.attlist "INCLUDE" >
<![%SVG.vkern.attlist;[
```

```
<!ATTLIST %SVG.vkern.qname;
  %SVG.Core.attrib;
  u1 CDATA #IMPLIED
  g1 CDATA #IMPLIED
  u2 CDATA #IMPLIED
  g2 CDATA #IMPLIED
  k %Number.datatype; #REQUIRED
>
<!-- end of SVG.vkern.attlist -->]]>

<!-- font-face-src: Font Face Source Element ..... -->

<!ENTITY % SVG.font-face-src.extra.content "" >

<!ENTITY % SVG.font-face-src.element "INCLUDE" >
<![%SVG.font-face-src.element;[
<!ENTITY % SVG.font-face-src.content
  "( %SVG.font-face-uri.qname; | %SVG.font-face-name.qname;
    %SVG.font-face-src.extra.content; )+"
>
<!ELEMENT %SVG.font-face-src.qname; %SVG.font-face-src.content; >
<!-- end of SVG.font-face-src.element -->]]>

<!ENTITY % SVG.font-face-src.attlist "INCLUDE" >
<![%SVG.font-face-src.attlist;[
<!ATTLIST %SVG.font-face-src.qname;
  %SVG.Core.attrib;
>
<!-- end of SVG.font-face-src.attlist -->]]>

<!-- font-face-uri: Font Face URI Element ..... -->

<!ENTITY % SVG.font-face-uri.extra.content "" >

<!ENTITY % SVG.font-face-uri.element "INCLUDE" >
<![%SVG.font-face-uri.element;[
<!ENTITY % SVG.font-face-uri.content
  "( %SVG.font-face-format.qname; %SVG.font-face-uri.extra.content; )*"
>
<!ELEMENT %SVG.font-face-uri.qname; %SVG.font-face-uri.content; >
<!-- end of SVG.font-face-uri.element -->]]>

<!ENTITY % SVG.font-face-uri.attlist "INCLUDE" >
<![%SVG.font-face-uri.attlist;[
<!ATTLIST %SVG.font-face-uri.qname;
  %SVG.Core.attrib;
  %SVG.XLinkRequired.attrib;
>
```

```
<!-- end of SVG.font-face-uri.attlist -->]]>

<!-- font-face-format: Font Face Format Element ..... -->

<!ENTITY % SVG.font-face-format.element "INCLUDE" >
<![%SVG.font-face-format.element;[
<!ENTITY % SVG.font-face-format.content "EMPTY" >
<!ELEMENT %SVG.font-face-format.qname; %SVG.font-face-format.content; >
<!-- end of SVG.font-face-format.element -->]]>

<!ENTITY % SVG.font-face-format.attlist "INCLUDE" >
<![%SVG.font-face-format.attlist;[
<!ATTLIST %SVG.font-face-format.qname;
  %SVG.Core.attrib;
  string CDATA #IMPLIED
>
<!-- end of SVG.font-face-format.attlist -->]]>

<!-- font-face-name: Font Face Name Element ..... -->

<!ENTITY % SVG.font-face-name.element "INCLUDE" >
<![%SVG.font-face-name.element;[
<!ENTITY % SVG.font-face-name.content "EMPTY" >
<!ELEMENT %SVG.font-face-name.qname; %SVG.font-face-name.content; >
<!-- end of SVG.font-face-name.element -->]]>

<!ENTITY % SVG.font-face-name.attlist "INCLUDE" >
<![%SVG.font-face-name.attlist;[
<!ATTLIST %SVG.font-face-name.qname;
  %SVG.Core.attrib;
  name CDATA #IMPLIED
>
<!-- end of SVG.font-face-name.attlist -->]]>

<!-- definition-src: Definition Source Element ..... -->

<!ENTITY % SVG.definition-src.element "INCLUDE" >
<![%SVG.definition-src.element;[
<!ENTITY % SVG.definition-src.content "EMPTY" >
<!ELEMENT %SVG.definition-src.qname; %SVG.definition-src.content; >
<!-- end of SVG.definition-src.element -->]]>

<!ENTITY % SVG.definition-src.attlist "INCLUDE" >
<![%SVG.definition-src.attlist;[
<!ATTLIST %SVG.definition-src.qname;
  %SVG.Core.attrib;
  %SVG.XLinkRequired.attrib;
>
```

<!-- end of SVG.definition-src.attlist -->]]>

<!-- end of svg-font.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-font.mod
No license file was found, but licenses were detected in source scan.

<!--
File isolat1.ent produced by the XSL script entities.xsl
from input data in unicode.xml.

Copyright 1998 - 2009 W3C.

Use and distribution of this code are permitted under the terms of the
W3C Software Notice and License.
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

Please report any errors to David Carlisle
via the public W3C list www-math@w3.org.

The numeric character values assigned to each entity
(should) match the Unicode assignments in Unicode 5.x.

Entity names in this file are derived from files carrying the
following notice:

(C) International Organization for Standardization 1986
Permission to copy in any form is granted for use with
conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Added Latin 1//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isolat1.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isolat1 PUBLIC
"-//W3C//ENTITIES Added Latin 1//EN//XML"

```
      "http://www.w3.org/2003/entities/2007/isolat1.ent"
    >
    %isolat1;

-->

<!ENTITY aacute       "&#x000E1;" ><!--LATIN SMALL LETTER A WITH ACUTE -->
<!ENTITY Aacute       "&#x000C1;" ><!--LATIN CAPITAL LETTER A WITH ACUTE -->
<!ENTITY acirc       "&#x000E2;" ><!--LATIN SMALL LETTER A WITH CIRCUMFLEX -->
<!ENTITY Acirc       "&#x000C2;" ><!--LATIN CAPITAL LETTER A WITH CIRCUMFLEX -->
<!ENTITY aelig       "&#x000E6;" ><!--LATIN SMALL LETTER AE -->
<!ENTITY AElig       "&#x000C6;" ><!--LATIN CAPITAL LETTER AE -->
<!ENTITY agrave       "&#x000E0;" ><!--LATIN SMALL LETTER A WITH GRAVE -->
<!ENTITY Agrave       "&#x000C0;" ><!--LATIN CAPITAL LETTER A WITH GRAVE -->
<!ENTITY aring       "&#x000E5;" ><!--LATIN SMALL LETTER A WITH RING ABOVE -->
<!ENTITY Aring       "&#x000C5;" ><!--LATIN CAPITAL LETTER A WITH RING ABOVE -->
<!ENTITY atilde       "&#x000E3;" ><!--LATIN SMALL LETTER A WITH TILDE -->
<!ENTITY Atilde       "&#x000C3;" ><!--LATIN CAPITAL LETTER A WITH TILDE -->
<!ENTITY auml       "&#x000E4;" ><!--LATIN SMALL LETTER A WITH DIAERESIS -->
<!ENTITY Auml       "&#x000C4;" ><!--LATIN CAPITAL LETTER A WITH DIAERESIS -->
<!ENTITY ccedil       "&#x000E7;" ><!--LATIN SMALL LETTER C WITH CEDILLA -->
<!ENTITY Ccedil       "&#x000C7;" ><!--LATIN CAPITAL LETTER C WITH CEDILLA -->
<!ENTITY eacute       "&#x000E9;" ><!--LATIN SMALL LETTER E WITH ACUTE -->
<!ENTITY Eacute       "&#x000C9;" ><!--LATIN CAPITAL LETTER E WITH ACUTE -->
<!ENTITY ecirc       "&#x000EA;" ><!--LATIN SMALL LETTER E WITH CIRCUMFLEX -->
<!ENTITY Ecirc       "&#x000CA;" ><!--LATIN CAPITAL LETTER E WITH CIRCUMFLEX -->
<!ENTITY egrave       "&#x000E8;" ><!--LATIN SMALL LETTER E WITH GRAVE -->
<!ENTITY Egrave       "&#x000C8;" ><!--LATIN CAPITAL LETTER E WITH GRAVE -->
<!ENTITY eth       "&#x000F0;" ><!--LATIN SMALL LETTER ETH -->
<!ENTITY ETH       "&#x000D0;" ><!--LATIN CAPITAL LETTER ETH -->
<!ENTITY euml       "&#x000EB;" ><!--LATIN SMALL LETTER E WITH DIAERESIS -->
<!ENTITY Euml       "&#x000CB;" ><!--LATIN CAPITAL LETTER E WITH DIAERESIS -->
<!ENTITY iacute       "&#x000ED;" ><!--LATIN SMALL LETTER I WITH ACUTE -->
<!ENTITY Iacute       "&#x000CD;" ><!--LATIN CAPITAL LETTER I WITH ACUTE -->
<!ENTITY icirc       "&#x000EE;" ><!--LATIN SMALL LETTER I WITH CIRCUMFLEX -->
<!ENTITY Icirc       "&#x000CE;" ><!--LATIN CAPITAL LETTER I WITH CIRCUMFLEX -->
<!ENTITY igrave       "&#x000EC;" ><!--LATIN SMALL LETTER I WITH GRAVE -->
<!ENTITY Igrave       "&#x000CC;" ><!--LATIN CAPITAL LETTER I WITH GRAVE -->
<!ENTITY iuml       "&#x000EF;" ><!--LATIN SMALL LETTER I WITH DIAERESIS -->
<!ENTITY Iuml       "&#x000CF;" ><!--LATIN CAPITAL LETTER I WITH DIAERESIS -->
<!ENTITY ntilde       "&#x000F1;" ><!--LATIN SMALL LETTER N WITH TILDE -->
<!ENTITY Ntilde       "&#x000D1;" ><!--LATIN CAPITAL LETTER N WITH TILDE -->
<!ENTITY oacute       "&#x000F3;" ><!--LATIN SMALL LETTER O WITH ACUTE -->
<!ENTITY Oacute       "&#x000D3;" ><!--LATIN CAPITAL LETTER O WITH ACUTE -->
<!ENTITY ocirc       "&#x000F4;" ><!--LATIN SMALL LETTER O WITH CIRCUMFLEX -->
<!ENTITY Ocirc       "&#x000D4;" ><!--LATIN CAPITAL LETTER O WITH CIRCUMFLEX -->
<!ENTITY ograve       "&#x000F2;" ><!--LATIN SMALL LETTER O WITH GRAVE -->
<!ENTITY Ograve       "&#x000D2;" ><!--LATIN CAPITAL LETTER O WITH GRAVE -->
```

```
<!ENTITY oslash         "&#x000F8;" ><!--LATIN SMALL LETTER O WITH STROKE -->
<!ENTITY Oslash         "&#x000D8;" ><!--LATIN CAPITAL LETTER O WITH STROKE -->
<!ENTITY otilde         "&#x000F5;" ><!--LATIN SMALL LETTER O WITH TILDE -->
<!ENTITY Otilde         "&#x000D5;" ><!--LATIN CAPITAL LETTER O WITH TILDE -->
<!ENTITY ouml           "&#x000F6;" ><!--LATIN SMALL LETTER O WITH DIAERESIS -->
<!ENTITY Ouml           "&#x000D6;" ><!--LATIN CAPITAL LETTER O WITH DIAERESIS -->
<!ENTITY szlig          "&#x000DF;" ><!--LATIN SMALL LETTER SHARP S -->
<!ENTITY thorn          "&#x000FE;" ><!--LATIN SMALL LETTER THORN -->
<!ENTITY THORN          "&#x000DE;" ><!--LATIN CAPITAL LETTER THORN -->
<!ENTITY uacute         "&#x000FA;" ><!--LATIN SMALL LETTER U WITH ACUTE -->
<!ENTITY Uacute         "&#x000DA;" ><!--LATIN CAPITAL LETTER U WITH ACUTE -->
<!ENTITY ucirc          "&#x000FB;" ><!--LATIN SMALL LETTER U WITH CIRCUMFLEX -->
<!ENTITY Ucirc          "&#x000DB;" ><!--LATIN CAPITAL LETTER U WITH CIRCUMFLEX -->
<!ENTITY ugrave         "&#x000F9;" ><!--LATIN SMALL LETTER U WITH GRAVE -->
<!ENTITY Ugrave         "&#x000D9;" ><!--LATIN CAPITAL LETTER U WITH GRAVE -->
<!ENTITY uuml           "&#x000FC;" ><!--LATIN SMALL LETTER U WITH DIAERESIS -->
<!ENTITY Uuml           "&#x000DC;" ><!--LATIN CAPITAL LETTER U WITH DIAERESIS -->
<!ENTITY yacute         "&#x000FD;" ><!--LATIN SMALL LETTER Y WITH ACUTE -->
<!ENTITY Yacute         "&#x000DD;" ><!--LATIN CAPITAL LETTER Y WITH ACUTE -->
<!ENTITY yuml           "&#x000FF;" ><!--LATIN SMALL LETTER Y WITH DIAERESIS -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isolat1.ent
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Inputmode Module  ................................................ -->
<!-- file: xhtml-inputmode-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2007 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-inputmode-1.mod,v 1.2 2007/07/13 14:37:53 jigsaw Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Inputmode 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-inputmode-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Inputmode

      inputmode

   This module declares the 'inputmode' attribute used for suggesting the
```

```
   input mode associated with an input or textarea element.
-->

<!-- render in this frame -->
<!ENTITY % Inputmode.datatype "CDATA" >

<!-- add 'inputmode' attribute to 'input' element -->
<!ATTLIST %input.qname;
    inputmode       %Inputmode.datatype;   #IMPLIED
>

<!-- add 'inputmode' attribute to 'textarea' element -->
<!ATTLIST %textarea.qname;
    inputmode       %Inputmode.datatype;   #IMPLIED
>

<!-- end of xhtml-inputmode-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-inputmode-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- XHTML Role Qname Module  ..... -->
<!-- file: xhtml-role-qname-1.mod

   This is XHTML Role - the Role Attribute Module for XHTML.

   Copyright 2006 W3C (MIT, ERCIM, Keio), All Rights Reserved.

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Role Attribute Qnames 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-role-qname-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- XHTML Role Attribute Qname (Qualified Name) Module

   This module is contained in two parts, labeled Section 'A' and 'B':

     Section A declares parameter entities to support namespace-
     qualified names, namespace declarations, and name prefixing
     for XHTML Role and extensions.
```

```
     Section B declares parameter entities used to provide
     namespace-qualified names for the XHTML role attribute:

      %role.qname;   the xmlns-qualified name for @role
      ...

   XHTML Role extensions would create a module similar to this one.
-->

<!-- Section A: XHTML Role Attribute XML Namespace Framework :::::::::::::: -->

<!-- 1. Declare a %XHTML-ROLE.prefixed; conditional section keyword, used
      to activate namespace prefixing. The default value should
      inherit '%NS.prefixed;' from the DTD driver, so that unless
      overridden, the default behavior follows the overall DTD
      prefixing scheme.
-->
<!ENTITY % NS.prefixed "IGNORE" >
<!ENTITY % XHTML-ROLE.prefixed "%NS.prefixed;" >

<!-- 2. Declare a parameter entity (eg., %XHTML-ROLE.xmlns;) containing
      the URI reference used to identify the XHTML Role Attribute namespace
-->
<!ENTITY % XHTML-ROLE.xmlns  "http://www.w3.org/1999/xhtml" >

<!-- 3. Declare parameter entities (eg., %XML.prefix;) containing
      the default namespace prefix string(s) to use when prefixing
      is enabled. This may be overridden in the DTD driver or the
      internal subset of an document instance. If no default prefix
      is desired, this may be declared as an empty string.

   NOTE: As specified in [XMLNAMES], the namespace prefix serves
   as a proxy for the URI reference, and is not in itself significant.
-->
<!ENTITY % XHTML-ROLE.prefix  "" >

<!-- 4. Declare parameter entities (eg., %XHTML-ROLE.pfx;) containing the
      colonized prefix(es) (eg., '%XHTML-ROLE.prefix;:') used when
      prefixing is active, an empty string when it is not.
-->
<![%XHTML-ROLE.prefixed;[
<!ENTITY % XHTML-ROLE.pfx  "%XHTML-ROLE.prefix;:" >
]]>
<!ENTITY % XHTML-ROLE.pfx  "" >

<!-- declare qualified name extensions here ..... -->
<!ENTITY % xhtml-role-qname-extra.mod "" >
%xhtml-role-qname-extra.mod;
```

```
<!-- 5. The parameter entity %XHTML-ROLE.xmlns.extra.attrib; may be
      redeclared to contain any non-XHTML Role Attribute namespace
      declaration attributes for namespaces embedded in XML. The default
      is an empty string.  XLink should be included here if used
      in the DTD.
-->
<!ENTITY % XHTML-ROLE.xmlns.extra.attrib "" >


<!-- Section B: XML Qualified Names ::::::::::::::::::::::::::::: -->

<!-- 6. This section declares parameter entities used to provide
      namespace-qualified names for the XHTML role attribute.
-->

<!ENTITY % xhtml-role.role.qname  "%XHTML-ROLE.pfx;role" >


<!-- The following defines a PE for use in the attribute sets of elements in
   other namespaces that want to incorporate the XML Event attributes. Note
   that in this case the XHTML-ROLE.pfx should always be defined. -->

<!ENTITY % xhtml-role.attrs.qname
  "%XHTML-ROLE.pfx;role            CDATA       #IMPLIED"
  >

<!-- end of xhtml-role-qname-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-role-qname-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Mask Module ..................................................... -->
<!-- file: svg-mask.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-mask.mod,v 1.4 2002/11/14 15:11:03 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Mask//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-mask.mod"

   ..................................................................... -->
```

```
<!-- Mask

     mask

   This module declares markup to provide support for masking.
-->

<!-- 'mask' property/attribute value (e.g., 'none', <uri>) -->
<!ENTITY % MaskValue.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.mask.qname "mask" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Mask.class ..... -->

<!ENTITY % SVG.Mask.extra.class "" >

<!ENTITY % SVG.Mask.class
  "| %SVG.mask.qname; %SVG.Mask.extra.class;"
>

<!-- SVG.Mask.attrib ..... -->
```

```
<!ENTITY % SVG.Mask.extra.attrib "" >

<!ENTITY % SVG.Mask.attrib
  "mask %MaskValue.datatype; #IMPLIED
   %SVG.Mask.extra.attrib;"
>

<!-- SVG.Presentation.attrib ..... -->

<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
   %SVG.Viewport.attrib;
   %SVG.Text.attrib;
   %SVG.TextContent.attrib;
   %SVG.Font.attrib;
   %SVG.Paint.attrib;
   %SVG.Color.attrib;
   %SVG.Opacity.attrib;
   %SVG.Graphics.attrib;
   %SVG.Marker.attrib;
   %SVG.ColorProfile.attrib;
   %SVG.Gradient.attrib;
   %SVG.Clip.attrib;
   %SVG.Mask.attrib;
   %SVG.Filter.attrib;
   %SVG.FilterColor.attrib;
   %SVG.Cursor.attrib;
   flood-color %SVGColor.datatype; #IMPLIED
   flood-opacity %OpacityValue.datatype; #IMPLIED
   lighting-color %SVGColor.datatype; #IMPLIED
   %SVG.Presentation.extra.attrib;"
>

<!-- mask: Mask Element ..... -->

<!ENTITY % SVG.mask.extra.content "" >

<!ENTITY % SVG.mask.element "INCLUDE" >
<![%SVG.mask.element;[
<!ENTITY % SVG.mask.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
     %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
     %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
     %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
     %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
     %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
```

```
    %SVG.Font.class; %SVG.mask.extra.content; )*"
>
<!ELEMENT %SVG.mask.qname; %SVG.mask.content; >
<!-- end of SVG.mask.element --]]>

<!ENTITY % SVG.mask.attlist "INCLUDE" >
<![%SVG.mask.attlist;[
<!ATTLIST %SVG.mask.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #IMPLIED
  height %Length.datatype; #IMPLIED
  maskUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
  maskContentUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
>
<!-- end of SVG.mask.attlist --]]>

<!-- end of svg-mask.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-mask.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Java Applet Module  ..... -->
<!-- file: xhtml-applet-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-applet-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Java Applets 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-applet-1.mod"

   Revisions:
#2001-02-20: added PCDATA to content model of applet
   ..... -->

<!-- Java Applets
```

```
      applet

   This module declares the applet element type and its attributes,
   used to provide support for Java applets. The 'alt' attribute is
   now required (as it is on images). One of either code or object
   attributes must be present. In the document, place param elements
   before other content within the <applet> element.

   Note that use of this module requires instantiation of the Param
   Element Module.
-->

<!-- applet: Java Applet ..... -->

<!ENTITY % applet.element  "INCLUDE" >
<![%applet.element;[
<!ENTITY % applet.content
   "( #PCDATA | %param.qname; | %Flow.mix; )*"
>
<!ENTITY % applet.qname  "applet" >
<!ELEMENT %applet.qname;  %applet.content; >
<!-- end of applet.element -->]]>

<!ENTITY % applet.attlist  "INCLUDE" >
<![%applet.attlist;[
<!ATTLIST %applet.qname;
    %Core.attrib;
    alt         %Text.datatype;         #REQUIRED
    archive      CDATA                   #IMPLIED
    code        CDATA                   #IMPLIED
    codebase     %URI.datatype;          #IMPLIED
    object      CDATA                   #IMPLIED
    width        %Length.datatype;       #REQUIRED
    height       %Length.datatype;       #REQUIRED
>
<!-- end of applet.attlist -->]]>

<!-- end of xhtml-applet-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-applet-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Access Qname Module  ..... -->
<!-- file: xhtml-access-qname-1.mod
```

```
    This is XHTML Access - the Access Attribute Module for XHTML.

    Copyright 2007-2008 W3C (MIT, ERCIM, Keio), All Rights Reserved.

    This DTD module is identified by the PUBLIC and SYSTEM identifiers:

      PUBLIC "-//W3C//ENTITIES XHTML Access Attribute Qnames 1.0//EN"
      SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-access-qname-1.mod"

    Revisions:
    (none)
    ..... -->

<!-- XHTML Access Attribute Qname (Qualified Name) Module

    This module is contained in two parts, labeled Section 'A' and 'B':

      Section A declares parameter entities to support namespace-
      qualified names, namespace declarations, and name prefixing
      for XHTML Access and extensions.

      Section B declares parameter entities used to provide
      namespace-qualified names for the XHTML access element:

       %XHTML-ACCESS.access.qname;   the xmlns-qualified name for access
       ...

    XHTML Access extensions would create a module similar to this one.
-->

<!-- Section A: XHTML Access Attribute XML Namespace Framework :::::::::::::::: -->

<!-- 1. Declare a %XHTML-ACCESS.prefixed; conditional section keyword, used
      to activate namespace prefixing. The default value should
      inherit '%NS.prefixed;' from the DTD driver, so that unless
      overridden, the default behavior follows the overall DTD
      prefixing scheme.
-->
<!ENTITY % NS.prefixed "IGNORE" >
<!ENTITY % XHTML-ACCESS.prefixed "%NS.prefixed;" >

<!-- 2. Declare a parameter entity (eg., %XHTML-ACCESS.xmlns;) containing
      the URI reference used to identify the XHTML Access Attribute namespace
-->
<!ENTITY % XHTML-ACCESS.xmlns  "http://www.w3.org/1999/xhtml" >

<!-- 3. Declare parameter entities (eg., %XML.prefix;) containing
      the default namespace prefix string(s) to use when prefixing
```

```
      is enabled. This may be overridden in the DTD driver or the
      internal subset of an document instance. If no default prefix
      is desired, this may be declared as an empty string.

    NOTE: As specified in [XMLNAMES], the namespace prefix serves
    as a proxy for the URI reference, and is not in itself significant.
-->
<!ENTITY % XHTML-ACCESS.prefix  "" >

<!-- 4. Declare parameter entities (eg., %XHTML-ACCESS.pfx;) containing the
      colonized prefix(es) (eg., '%XHTML-ACCESS.prefix;:') used when
      prefixing is active, an empty string when it is not.
-->
<![%XHTML-ACCESS.prefixed;[
<!ENTITY % XHTML-ACCESS.pfx  "%XHTML-ACCESS.prefix;:" >
]]>
<!ENTITY % XHTML-ACCESS.pfx  "" >

<!-- declare qualified name extensions here ............ -->
<!ENTITY % xhtml-access-qname-extra.mod "" >
%xhtml-access-qname-extra.mod;

<!-- 5. The parameter entity %XHTML-ACCESS.xmlns.extra.attrib; may be
      redeclared to contain any non-XHTML Access namespace
      declaration attributes for namespaces embedded in XML. The default
      is an empty string.  XLink should be included here if used
      in the DTD.
-->
<!ENTITY % XHTML-ACCESS.xmlns.extra.attrib "" >


<!-- Section B: XML Qualified Names ::::::::::::::::::::::::::::: -->

<!-- 6. This section declares parameter entities used to provide
      namespace-qualified names for the XHTML Access element.
-->

<!ENTITY % XHTML-ACCESS.access.qname  "%XHTML-ACCESS.pfx;access" >

<!-- end of xhtml-access-qname-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-access-qname-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Animation Events Attribute Module ........................... -->
```

```
<!-- file: svg-animevents-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-animevents-attrib.mod,v 1.3 2002/04/28 13:50:23 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Animation Events Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-animevents-attrib.mod"

   ..................................................................... -->

<!-- Animation Events Attribute

      onbegin, onend, onrepeat, onload

   This module defines the AnimationEvents attribute set.
-->

<!ENTITY % SVG.onbegin.attrib
  "onbegin %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onend.attrib
  "onend %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onrepeat.attrib
  "onrepeat %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onload.attrib
  "onload %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.AnimationEvents.extra.attrib "" >

<!ENTITY % SVG.AnimationEvents.attrib
  "%SVG.onbegin.attrib;
   %SVG.onend.attrib;
   %SVG.onrepeat.attrib;
   %SVG.onload.attrib;
   %SVG.AnimationEvents.extra.attrib;"
>

<!-- end of svg-animevents-attrib.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-animevents-attrib.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Param Element Module  ..................................... -->
<!-- file: xhtml-param-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-param-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Param Element 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-param-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Parameters for Java Applets and Embedded Objects

      param

   This module provides declarations for the param element,
   used to provide named property values for the applet
   and object elements.
-->

<!-- param: Named Property Value ..................... -->

<!ENTITY % param.element  "INCLUDE" >
<![%param.element;[
<!ENTITY % param.content  "EMPTY" >
<!ENTITY % param.qname  "param" >
<!ELEMENT %param.qname;  %param.content; >
<!-- end of param.element -->]]>

<!ENTITY % param.attlist  "INCLUDE" >
<![%param.attlist;[
<!ATTLIST %param.qname;
    %XHTML.xmlns.attrib;
    %id.attrib;
    name        CDATA                #REQUIRED
    value       CDATA                #IMPLIED
    valuetype    ( data | ref | object )  'data'
```

```
    type        %ContentType.datatype;   #IMPLIED
>
<!-- end of param.attlist -->]]>

<!-- end of xhtml-param-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-param-1.mod
No license file was found, but licenses were detected in source scan.

```
20010904/

 Copyright (c) 2000 W3C (MIT, INRIA, Keio), All Rights Reserved.

 For SVG 1.0:

   Namespace:
     http://www.w3.org/2000/svg

   Public identifier:
     PUBLIC "
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg10/svg10.dtd
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Shape Module ................................................ -->
<!-- file: svg-shape.mod

    This is SVG, a language for describing two-dimensional graphics in XML.
    Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
    Revision: $Id: svg-shape.mod,v 1.3 2002/10/24 17:40:16 fujisawa Exp $

    This DTD module is identified by the PUBLIC and SYSTEM identifiers:

      PUBLIC "-//W3C//ELEMENTS SVG 1.1 Shape//EN"
      SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-shape.mod"

    ..................................................................... -->

<!-- Shape

      path, rect, circle, line, ellipse, polyline, polygon

    This module declares markup to provide support for graphical shapes.
```

```
-->

<!-- a list of points -->
<!ENTITY % Points.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.path.qname "path" >
<!ENTITY % SVG.rect.qname "rect" >
<!ENTITY % SVG.circle.qname "circle" >
<!ENTITY % SVG.line.qname "line" >
<!ENTITY % SVG.ellipse.qname "ellipse" >
<!ENTITY % SVG.polyline.qname "polyline" >
<!ENTITY % SVG.polygon.qname "polygon" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Shape.class ..... -->

<!ENTITY % SVG.Shape.extra.class "" >

<!ENTITY % SVG.Shape.class
   "| %SVG.path.qname; | %SVG.rect.qname; | %SVG.circle.qname;
   | %SVG.line.qname; | %SVG.ellipse.qname; | %SVG.polyline.qname;
   | %SVG.polygon.qname; %SVG.Shape.extra.class;"
>

<!-- path: Path Element ..... -->

<!ENTITY % SVG.path.extra.content "" >

<!ENTITY % SVG.path.element "INCLUDE" >
<![%SVG.path.element;[
```

```
<!ENTITY % SVG.path.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class;
     %SVG.path.extra.content; )*)"
>
<!ELEMENT %SVG.path.qname; %SVG.path.content; >
<!-- end of SVG.path.element -->]]>

<!ENTITY % SVG.path.attlist "INCLUDE" >
<![%SVG.path.attlist;[
<!ATTLIST %SVG.path.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Marker.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  d %PathData.datatype; #REQUIRED
  pathLength %Number.datatype; #IMPLIED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.path.attlist -->]]>

<!-- rect: Rectangle Element ..... -->

<!ENTITY % SVG.rect.extra.content "" >

<!ENTITY % SVG.rect.element "INCLUDE" >
<![%SVG.rect.element;[
<!ENTITY % SVG.rect.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class;
     %SVG.rect.extra.content; )*)"
>
<!ELEMENT %SVG.rect.qname; %SVG.rect.content; >
<!-- end of SVG.rect.element -->]]>

<!ENTITY % SVG.rect.attlist "INCLUDE" >
<![%SVG.rect.attlist;[
<!ATTLIST %SVG.rect.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
```

```
  %SVG.Style.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #REQUIRED
  height %Length.datatype; #REQUIRED
  rx %Length.datatype; #IMPLIED
  ry %Length.datatype; #IMPLIED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.rect.attlist -->]]>

<!-- circle: Circle Element ..... -->

<!ENTITY % SVG.circle.extra.content "" >

<!ENTITY % SVG.circle.element "INCLUDE" >
<![%SVG.circle.element;[
<!ENTITY % SVG.circle.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class;
     %SVG.circle.extra.content; )*)"
>
<!ELEMENT %SVG.circle.qname; %SVG.circle.content; >
<!-- end of SVG.circle.element -->]]>

<!ENTITY % SVG.circle.attlist "INCLUDE" >
<![%SVG.circle.attlist;[
<!ATTLIST %SVG.circle.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
```

```
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  cx %Coordinate.datatype; #IMPLIED
  cy %Coordinate.datatype; #IMPLIED
  r %Length.datatype; #REQUIRED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.circle.attlist -->]]>

<!-- line: Line Element ..... -->

<!ENTITY % SVG.line.extra.content "" >

<!ENTITY % SVG.line.element "INCLUDE" >
<![%SVG.line.element;[
<!ENTITY % SVG.line.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class;
     %SVG.line.extra.content; )*)"
>
<!ELEMENT %SVG.line.qname; %SVG.line.content; >
<!-- end of SVG.line.element -->]]>

<!ENTITY % SVG.line.attlist "INCLUDE" >
<![%SVG.line.attlist;[
<!ATTLIST %SVG.line.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Marker.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  x1 %Coordinate.datatype; #IMPLIED
  y1 %Coordinate.datatype; #IMPLIED
  x2 %Coordinate.datatype; #IMPLIED
  y2 %Coordinate.datatype; #IMPLIED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.line.attlist -->]]>

<!-- ellipse: Ellipse Element ..... -->
```

```
<!ENTITY % SVG.ellipse.extra.content "" >

<!ENTITY % SVG.ellipse.element "INCLUDE" >
<![%SVG.ellipse.element;[
<!ENTITY % SVG.ellipse.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class;
     %SVG.ellipse.extra.content; )*)"
>
<!ELEMENT %SVG.ellipse.qname; %SVG.ellipse.content; >
<!-- end of SVG.ellipse.element -->]]>

<!ENTITY % SVG.ellipse.attlist "INCLUDE" >
<![%SVG.ellipse.attlist;[
<!ATTLIST %SVG.ellipse.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  cx %Coordinate.datatype; #IMPLIED
  cy %Coordinate.datatype; #IMPLIED
  rx %Length.datatype; #REQUIRED
  ry %Length.datatype; #REQUIRED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.ellipse.attlist -->]]>

<!-- polyline: Polyline Element ..... -->

<!ENTITY % SVG.polyline.extra.content "" >

<!ENTITY % SVG.polyline.element "INCLUDE" >
<![%SVG.polyline.element;[
<!ENTITY % SVG.polyline.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class;
     %SVG.polyline.extra.content; )*)"
>
<!ELEMENT %SVG.polyline.qname; %SVG.polyline.content; >
<!-- end of SVG.polyline.element -->]]>
```

```
<!ENTITY % SVG.polyline.attlist "INCLUDE" >
<![%SVG.polyline.attlist;[
<!ATTLIST %SVG.polyline.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Marker.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  points %Points.datatype; #REQUIRED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.polyline.attlist -->]]>

<!-- polygon: Polygon Element ..... -->

<!ENTITY % SVG.polygon.extra.content "" >

<!ENTITY % SVG.polygon.element "INCLUDE" >
<![%SVG.polygon.element;[
<!ENTITY % SVG.polygon.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class;
     %SVG.polygon.extra.content; )*)"
>
<!ELEMENT %SVG.polygon.qname; %SVG.polygon.content; >
<!-- end of SVG.polygon.element -->]]>

<!ENTITY % SVG.polygon.attlist "INCLUDE" >
<![%SVG.polygon.attlist;[
<!ATTLIST %SVG.polygon.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Marker.attrib;
  %SVG.Clip.attrib;
```

```
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  points %Points.datatype; #REQUIRED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.polygon.attlist -->]]>

<!-- end of svg-shape.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-shape.mod
No license file was found, but licenses were detected in source scan.

```
<!--
   * This is a schema for the XML representation of the result of the
   * function fn:analyze-string()
   *
   * The schema is made available under the terms of the W3C software notice and license
   * at http://www.w3.org/Consortium/Legal/copyright-software-19980720
   *
  -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/analyze-string.xsd
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Style Module ................................................ -->
<!-- file: svg-style.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-style.mod,v 1.3 2002/10/24 17:40:16 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Style//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-style.mod"

   ..................................................................... -->

<!-- Style
```

```
      style

   This module declares markup to provide support for stylesheet.
-->

<!-- list of classes -->
<!ENTITY % ClassList.datatype "CDATA" >

<!-- comma-separated list of media descriptors. -->
<!ENTITY % MediaDesc.datatype "CDATA" >

<!-- style sheet data -->
<!ENTITY % StyleSheet.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.style.qname "style" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >

<!-- SVG.Style.class ..... -->

<!ENTITY % SVG.Style.extra.class "" >

<!ENTITY % SVG.Style.class
  "| %SVG.style.qname; %SVG.Style.extra.class;"
>

<!-- SVG.Style.attrib ..... -->

<!ENTITY % SVG.Style.extra.attrib "" >

<!ENTITY % SVG.Style.attrib
  "style %StyleSheet.datatype; #IMPLIED
   class %ClassList.datatype; #IMPLIED
   %SVG.Style.extra.attrib;"
>

<!-- style: Style Element ..... -->

<!ENTITY % SVG.style.extra.content "" >

<!ENTITY % SVG.style.element "INCLUDE" >
<![%SVG.style.element;[
<!ENTITY % SVG.style.content
  "( #PCDATA %SVG.style.extra.content; )*"
```

```
>
<!ELEMENT %SVG.style.qname; %SVG.style.content; >
<!-- end of SVG.style.element -->]]>

<!ENTITY % SVG.style.attlist "INCLUDE" >
<![%SVG.style.attlist;[
<!ATTLIST %SVG.style.qname;
  xml:space ( preserve ) #FIXED 'preserve'
  %SVG.Core.attrib;
  type %ContentType.datatype; #REQUIRED
  media %MediaDesc.datatype; #IMPLIED
  title %Text.datatype; #IMPLIED
>
<!-- end of SVG.style.attlist -->]]>

<!-- end of svg-style.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-style.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Extensibility Module ......................................... -->
<!-- file: svg-extensibility.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-extensibility.mod,v 1.4 2002/11/14 15:11:02 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Extensibility//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-extensibility.mod"

   ..................................................................... -->

<!-- Extensibility

     foreignObject

   This module declares markup to provide support for extensibility.
-->

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.foreignObject.qname "foreignObject" >
```

```
<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Extensibility.class ..... -->

<!ENTITY % SVG.Extensibility.extra.class "" >

<!ENTITY % SVG.Extensibility.class
  "| %SVG.foreignObject.qname; %SVG.Extensibility.extra.class;"
>

<!-- SVG.Presentation.attrib ..... -->

<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
   %SVG.Viewport.attrib;
   %SVG.Text.attrib;
   %SVG.TextContent.attrib;
   %SVG.Font.attrib;
   %SVG.Paint.attrib;
   %SVG.Color.attrib;
   %SVG.Opacity.attrib;
   %SVG.Graphics.attrib;
   %SVG.Marker.attrib;
   %SVG.ColorProfile.attrib;
   %SVG.Gradient.attrib;
```

```
   %SVG.Clip.attrib;
   %SVG.Mask.attrib;
   %SVG.Filter.attrib;
   %SVG.FilterColor.attrib;
   %SVG.Cursor.attrib;
   flood-color %SVGColor.datatype; #IMPLIED
   flood-opacity %OpacityValue.datatype; #IMPLIED
   lighting-color %SVGColor.datatype; #IMPLIED
   %SVG.Presentation.extra.attrib;"
>

<!-- foreignObject: Foreign Object Element ..... -->

<!ENTITY % SVG.foreignObject.extra.content "" >

<!ENTITY % SVG.foreignObject.element "INCLUDE" >
<![%SVG.foreignObject.element;[
<!ENTITY % SVG.foreignObject.content
  "( #PCDATA %SVG.foreignObject.extra.content; )*"
>
<!ELEMENT %SVG.foreignObject.qname; %SVG.foreignObject.content; >
<!-- end of SVG.foreignObject.element -->]]>

<!ENTITY % SVG.foreignObject.attlist "INCLUDE" >
<![%SVG.foreignObject.attlist;[
<!ATTLIST %SVG.foreignObject.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #REQUIRED
  height %Length.datatype; #REQUIRED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.foreignObject.attlist -->]]>

<!-- end of svg-extensibility.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-extensibility.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
```

<!-- XHTML Role Module ..... -->
<!-- file: xhtml-role-1.mod

This is XHTML Role - the Role Attribute Module for XHTML.

Copyright 2006 W3C (MIT, ERCIM, Keio), All Rights Reserved.

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES XHTML Role Attribute 1.0//EN"
SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-role-1.mod"

Revisions:
(none)
..... -->

<!-- XHTML Role Attribute Module does nothing, since role is not incorporated
in any elements content model as part of this module. -->
<!-- end of xhtml-role-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-role-1.mod
No license file was found, but licenses were detected in source scan.

2002 W3C (MIT, INRIA, Keio),
All Rights Reserved.

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml10/xhtml1-frameset.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Scripting Module ..... -->
<!-- file: svg-script.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-script.mod,v 1.3 2002/10/24 17:40:16 fujisawa Exp $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

```
      PUBLIC "-//W3C//ELEMENTS SVG 1.1 Scripting//EN"
      SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-script.mod"

    ..................................................................... -->

<!-- Scripting

      script

    This module declares markup to provide support for scripting.
-->

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.script.qname "script" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Script.class ..... -->

<!ENTITY % SVG.Script.extra.class "" >

<!ENTITY % SVG.Script.class
   "| %SVG.script.qname; %SVG.Script.extra.class;"
>

<!-- script: Script Element ..... -->

<!ENTITY % SVG.script.extra.content "" >

<!ENTITY % SVG.script.element "INCLUDE" >
<![%SVG.script.element;[
<!ENTITY % SVG.script.content
   "( #PCDATA %SVG.script.extra.content; )*"
>
<!ELEMENT %SVG.script.qname; %SVG.script.content; >
<!-- end of SVG.script.element -->]]>

<!ENTITY % SVG.script.attlist "INCLUDE" >
<![%SVG.script.attlist;[
<!ATTLIST %SVG.script.qname;
   %SVG.Core.attrib;
   %SVG.XLink.attrib;
   %SVG.External.attrib;
```

```
  type %ContentType.datatype; #REQUIRED
>
<!-- end of SVG.script.attlist -->]]>

<!-- end of svg-script.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-script.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Block Structural Module  ..................................... -->
<!-- file: xhtml-blkstruct-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-blkstruct-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Block Structural 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-blkstruct-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Block Structural

      div, p

   This module declares the elements and their attributes used to
   support block-level structural markup.
-->

<!ENTITY % div.element  "INCLUDE" >
<![%div.element;[
<!ENTITY % div.content
   "( #PCDATA | %Flow.mix; )*"
>
<!ENTITY % div.qname  "div" >
<!ELEMENT %div.qname;  %div.content; >
<!-- end of div.element -->]]>

<!ENTITY % div.attlist  "INCLUDE" >
<![%div.attlist;[
<!ATTLIST %div.qname;
```

```
    %Common.attrib;
>
<!-- end of div.attlist -->]]>

<!ENTITY % p.element  "INCLUDE" >
<![%p.element;[
<!ENTITY % p.content
   "( #PCDATA | %Inline.mix; )*" >
<!ENTITY % p.qname  "p" >
<!ELEMENT %p.qname;  %p.content; >
<!-- end of p.element -->]]>

<!ENTITY % p.attlist  "INCLUDE" >
<![%p.attlist;[
<!ATTLIST %p.qname;
    %Common.attrib;
>
<!-- end of p.attlist -->]]>

<!-- end of xhtml-blkstruct-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-blkstruct-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Lists Module  ..... -->
<!-- file: xhtml-list-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-list-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Lists 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-list-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Lists

      dl, dt, dd, ol, ul, li

   This module declares the list-oriented element types
```

```
   and their attributes.
-->

<!ENTITY % dl.qname  "dl" >
<!ENTITY % dt.qname  "dt" >
<!ENTITY % dd.qname  "dd" >
<!ENTITY % ol.qname  "ol" >
<!ENTITY % ul.qname  "ul" >
<!ENTITY % li.qname  "li" >

<!-- dl: Definition List ..... -->

<!ENTITY % dl.element  "INCLUDE" >
<![%dl.element;[
<!ENTITY % dl.content  "( %dt.qname; | %dd.qname; )+" >
<!ELEMENT %dl.qname;  %dl.content; >
<!-- end of dl.element -->]]>

<!ENTITY % dl.attlist  "INCLUDE" >
<![%dl.attlist;[
<!ATTLIST %dl.qname;
    %Common.attrib;
>
<!-- end of dl.attlist -->]]>

<!-- dt: Definition Term ..... -->

<!ENTITY % dt.element  "INCLUDE" >
<![%dt.element;[
<!ENTITY % dt.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ELEMENT %dt.qname;  %dt.content; >
<!-- end of dt.element -->]]>

<!ENTITY % dt.attlist  "INCLUDE" >
<![%dt.attlist;[
<!ATTLIST %dt.qname;
    %Common.attrib;
>
<!-- end of dt.attlist -->]]>

<!-- dd: Definition Description ..... -->

<!ENTITY % dd.element  "INCLUDE" >
<![%dd.element;[
<!ENTITY % dd.content
   "( #PCDATA | %Flow.mix; )*"
```

```
>
<!ELEMENT %dd.qname;  %dd.content; >
<!-- end of dd.element -->]]>

<!ENTITY % dd.attlist  "INCLUDE" >
<![%dd.attlist;[
<!ATTLIST %dd.qname;
    %Common.attrib;
>
<!-- end of dd.attlist -->]]>

<!-- ol: Ordered List (numbered styles) ..... -->

<!ENTITY % ol.element  "INCLUDE" >
<![%ol.element;[
<!ENTITY % ol.content  "( %li.qname; )+" >
<!ELEMENT %ol.qname;  %ol.content; >
<!-- end of ol.element -->]]>

<!ENTITY % ol.attlist  "INCLUDE" >
<![%ol.attlist;[
<!ATTLIST %ol.qname;
    %Common.attrib;
>
<!-- end of ol.attlist -->]]>

<!-- ul: Unordered List (bullet styles) ..... -->

<!ENTITY % ul.element  "INCLUDE" >
<![%ul.element;[
<!ENTITY % ul.content  "( %li.qname; )+" >
<!ELEMENT %ul.qname;  %ul.content; >
<!-- end of ul.element -->]]>

<!ENTITY % ul.attlist  "INCLUDE" >
<![%ul.attlist;[
<!ATTLIST %ul.qname;
    %Common.attrib;
>
<!-- end of ul.attlist -->]]>

<!-- li: List Item ..... -->

<!ENTITY % li.element  "INCLUDE" >
<![%li.element;[
<!ENTITY % li.content
   "( #PCDATA | %Flow.mix; )*"
>
```

```
<!ELEMENT %li.qname;  %li.content; >
<!-- end of li.element -->]]>

<!ENTITY % li.attlist  "INCLUDE" >
<![%li.attlist;[
<!ATTLIST %li.qname;
    %Common.attrib;
>
<!-- end of li.attlist -->]]>

<!-- end of xhtml-list-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-list-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML RDFa Qname Module  ..... -->
<!-- file: xhtml-rdfa-qname-1.mod

   This is XHTML RDFa - the RDFa Attribute Module for XHTML.

   Copyright 2007 W3C (MIT, ERCIM, Keio), All Rights Reserved.

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML RDFa Attribute Qnames 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-qname-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- XHTML RDFa Attribute Qname (Qualified Name) Module

   This module is contained in two parts, labeled Section 'A' and 'B':

     Section A declares parameter entities to support namespace-
     qualified names, namespace declarations, and name prefixing
     for XHTML RDFa and extensions.

     Section B declares parameter entities used to provide
     namespace-qualified names for the XHTML RDFa elements
     and attributes:

       %link.qname;   the xmlns-qualified name for link
       ...
```

```
   XHTML RDFa extensions would create a module similar to this one.
-->

<!-- Section A: XHTML RDFa Attribute XML Namespace Framework :::::::::::::: -->

<!-- 1. Declare a %XHTML-RDFA.prefixed; conditional section keyword, used
      to activate namespace prefixing. The default value should
      inherit '%NS.prefixed;' from the DTD driver, so that unless
      overridden, the default behavior follows the overall DTD
      prefixing scheme.
-->
<!ENTITY % NS.prefixed "IGNORE" >
<!ENTITY % XHTML-RDFA.prefixed "%NS.prefixed;" >

<!-- 2. Declare a parameter entity (eg., %XHTML-RDFA.xmlns;) containing
      the URI reference used to identify the XHTML RDFa Attribute namespace
-->
<!ENTITY % XHTML-RDFA.xmlns  "http://www.w3.org/1999/xhtml" >

<!-- 3. Declare parameter entities (eg., %XML.prefix;) containing
      the default namespace prefix string(s) to use when prefixing
      is enabled. This may be overridden in the DTD driver or the
      internal subset of an document instance. If no default prefix
      is desired, this may be declared as an empty string.

   NOTE: As specified in [XMLNAMES], the namespace prefix serves
   as a proxy for the URI reference, and is not in itself significant.
-->
<!ENTITY % XHTML-RDFA.prefix  "" >

<!-- 4. Declare parameter entities (eg., %XHTML-RDFA.pfx;) containing the
      colonized prefix(es) (eg., '%XHTML-RDFA.prefix;:') used when
      prefixing is active, an empty string when it is not.
-->
<![%XHTML-RDFA.prefixed;[
<!ENTITY % XHTML-RDFA.pfx  "%XHTML-RDFA.prefix;:" >
]]>
<![%XHTML.prefixed;[
<!ENTITY % XHTML-RDFA.pfx  "%XHTML.prefix;:" >
]]>
<!ENTITY % XHTML-RDFA.pfx  "" >

<!-- declare qualified name extensions here ..... -->
<!ENTITY % xhtml-rdfa-qname-extra.mod "" >
%xhtml-rdfa-qname-extra.mod;

<!-- 5. The parameter entity %XHTML-RDFA.xmlns.extra.attrib; may be
```

redeclared to contain any non-XHTML RDFa Attribute namespace
declaration attributes for namespaces embedded in XML. The default
is an empty string. XLink should be included here if used
in the DTD.
-->
<!ENTITY % XHTML-RDFA.xmlns.extra.attrib "" >

<!-- Section B: XML Qualified Names ::::::::::::::::::::::::::::: -->

<!-- 6. This section declares parameter entities used to provide
namespace-qualified names for the XHTML RDFa attribute modules.

Note that these names are NOT prefixed to be compatible with
XHTML Modularization 1.1
-->

<!-- end of xhtml-rdfa-qname-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-rdfa-qname-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Text Module ..... -->
<!-- file: svg-text.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-text.mod,v 1.4 2002/10/24 17:40:16 fujisawa Exp $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 Text//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-text.mod"

..... -->

<!-- Text

text, tspan, tref, textPath, altGlyph, altGlyphDef, altGlyphItem,
glyphRef

This module declares markup to provide support for alternate glyph.
-->

<!-- 'baseline-shift' property/attribute value (e.g., 'baseline', 'sub', etc.) -->

```
<!ENTITY % BaselineShiftValue.datatype "CDATA" >

<!-- 'font-family' property/attribute value (i.e., list of fonts) -->
<!ENTITY % FontFamilyValue.datatype "CDATA" >

<!-- 'font-size' property/attribute value -->
<!ENTITY % FontSizeValue.datatype "CDATA" >

<!-- 'font-size-adjust' property/attribute value -->
<!ENTITY % FontSizeAdjustValue.datatype "CDATA" >

<!-- 'glyph-orientation-horizontal' property/attribute value (e.g., <angle>) -->
<!ENTITY % GlyphOrientationHorizontalValue.datatype "CDATA" >

<!-- 'glyph-orientation-vertical' property/attribute value (e.g., 'auto', <angle>) -->
<!ENTITY % GlyphOrientationVerticalValue.datatype "CDATA" >

<!-- 'kerning' property/attribute value (e.g., 'auto', <length>) -->
<!ENTITY % KerningValue.datatype "CDATA" >

<!-- 'letter-spacing' or 'word-spacing' property/attribute value (e.g., 'normal', <length>) -->
<!ENTITY % SpacingValue.datatype "CDATA" >

<!-- 'text-decoration' property/attribute value (e.g., 'none', 'underline') -->
<!ENTITY % TextDecorationValue.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.text.qname "text" >
<!ENTITY % SVG.tspan.qname "tspan" >
<!ENTITY % SVG.tref.qname "tref" >
<!ENTITY % SVG.textPath.qname "textPath" >
<!ENTITY % SVG.altGlyph.qname "altGlyph" >
<!ENTITY % SVG.altGlyphDef.qname "altGlyphDef" >
<!ENTITY % SVG.altGlyphItem.qname "altGlyphItem" >
<!ENTITY % SVG.glyphRef.qname "glyphRef" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
```

```
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.XLinkRequired.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Text.class ..................................... -->

<!ENTITY % SVG.Text.extra.class "" >

<!ENTITY % SVG.Text.class
  "| %SVG.text.qname; | %SVG.altGlyphDef.qname; %SVG.Text.extra.class;"
>

<!-- SVG.TextContent.class ............................. -->

<!ENTITY % SVG.TextContent.extra.class "" >

<!ENTITY % SVG.TextContent.class
  "| %SVG.tspan.qname; | %SVG.tref.qname; | %SVG.textPath.qname;
   | %SVG.altGlyph.qname; %SVG.TextContent.extra.class;"
>

<!-- SVG.Text.attrib ..................................... -->

<!ENTITY % SVG.Text.extra.attrib "" >

<!ENTITY % SVG.Text.attrib
  "writing-mode ( lr-tb | rl-tb | tb-rl | lr | rl | tb | inherit ) #IMPLIED
   %SVG.Text.extra.attrib;"
>

<!-- SVG.TextContent.attrib ............................. -->

<!ENTITY % SVG.TextContent.extra.attrib "" >

<!ENTITY % SVG.TextContent.attrib
  "alignment-baseline ( auto | baseline | before-edge | text-before-edge |
                        middle | central | after-edge | text-after-edge |
                        ideographic | alphabetic | hanging | mathematical |
                        inherit ) #IMPLIED
   baseline-shift %BaselineShiftValue.datatype; #IMPLIED
   direction ( ltr | rtl | inherit ) #IMPLIED
   dominant-baseline ( auto | use-script | no-change | reset-size |
                       ideographic | alphabetic | hanging | mathematical |
                       central | middle | text-after-edge | text-before-edge |
                       inherit ) #IMPLIED
```

```
    glyph-orientation-horizontal %GlyphOrientationHorizontalValue.datatype;
                          #IMPLIED
    glyph-orientation-vertical %GlyphOrientationVerticalValue.datatype;
                          #IMPLIED
    kerning %KerningValue.datatype; #IMPLIED
    letter-spacing %SpacingValue.datatype; #IMPLIED
    text-anchor ( start | middle | end | inherit ) #IMPLIED
    text-decoration %TextDecorationValue.datatype; #IMPLIED
    unicode-bidi ( normal | embed | bidi-override | inherit ) #IMPLIED
    word-spacing %SpacingValue.datatype; #IMPLIED
    %SVG.TextContent.extra.attrib;"
>

<!-- SVG.Font.attrib ..................................... -->

<!ENTITY % SVG.Font.extra.attrib "" >

<!ENTITY % SVG.Font.attrib
  "font-family %FontFamilyValue.datatype; #IMPLIED
    font-size %FontSizeValue.datatype; #IMPLIED
    font-size-adjust %FontSizeAdjustValue.datatype; #IMPLIED
    font-stretch ( normal | wider | narrower | ultra-condensed |
                   extra-condensed | condensed | semi-condensed |
                   semi-expanded | expanded | extra-expanded |
                   ultra-expanded | inherit ) #IMPLIED
    font-style ( normal | italic | oblique | inherit ) #IMPLIED
    font-variant ( normal | small-caps | inherit ) #IMPLIED
    font-weight ( normal | bold | bolder | lighter | 100 | 200 | 300 | 400 |
                  500 | 600 | 700 | 800 | 900 | inherit ) #IMPLIED
    %SVG.Font.extra.attrib;"
>

<!-- text: Text Element ................................. -->

<!ENTITY % SVG.text.extra.content "" >

<!ENTITY % SVG.text.element "INCLUDE" >
<![%SVG.text.element;[
<!ENTITY % SVG.text.content
  "( #PCDATA | %SVG.Description.class; | %SVG.Animation.class;
     %SVG.TextContent.class; %SVG.Hyperlink.class;
     %SVG.text.extra.content; )*"
>
<!ELEMENT %SVG.text.qname; %SVG.text.content; >
<!-- end of SVG.text.element -->]]>

<!ENTITY % SVG.text.attlist "INCLUDE" >
<![%SVG.text.attlist;[
```

```
<!ATTLIST %SVG.text.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Text.attrib;
  %SVG.TextContent.attrib;
  %SVG.Font.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  x %Coordinates.datatype; #IMPLIED
  y %Coordinates.datatype; #IMPLIED
  dx %Lengths.datatype; #IMPLIED
  dy %Lengths.datatype; #IMPLIED
  rotate %Numbers.datatype; #IMPLIED
  textLength %Length.datatype; #IMPLIED
  lengthAdjust ( spacing | spacingAndGlyphs ) #IMPLIED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.text.attlist -->]]>

<!-- tspan: Text Span Element ..... -->

<!ENTITY % SVG.tspan.extra.content "" >

<!ENTITY % SVG.tspan.element "INCLUDE" >
<![%SVG.tspan.element;[
<!ENTITY % SVG.tspan.content
  "( #PCDATA | %SVG.tspan.qname; | %SVG.tref.qname; | %SVG.altGlyph.qname;
   | %SVG.animate.qname; | %SVG.set.qname; | %SVG.animateColor.qname;
   | %SVG.Description.class; %SVG.Hyperlink.class;
     %SVG.tspan.extra.content; )*"
>
<!ELEMENT %SVG.tspan.qname; %SVG.tspan.content; >
<!-- end of SVG.tspan.element -->]]>

<!ENTITY % SVG.tspan.attlist "INCLUDE" >
<![%SVG.tspan.attlist;[
<!ATTLIST %SVG.tspan.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
```

```
  %SVG.Style.attrib;
  %SVG.TextContent.attrib;
  %SVG.Font.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  x %Coordinates.datatype; #IMPLIED
  y %Coordinates.datatype; #IMPLIED
  dx %Lengths.datatype; #IMPLIED
  dy %Lengths.datatype; #IMPLIED
  rotate %Numbers.datatype; #IMPLIED
  textLength %Length.datatype; #IMPLIED
  lengthAdjust ( spacing | spacingAndGlyphs ) #IMPLIED
>
<!-- end of SVG.tspan.attlist -->]]>

<!-- tref: Text Reference Element ..... -->

<!ENTITY % SVG.tref.extra.content "" >

<!ENTITY % SVG.tref.element "INCLUDE" >
<![%SVG.tref.element;[
<!ENTITY % SVG.tref.content
  "( %SVG.animate.qname; | %SVG.set.qname; | %SVG.animateColor.qname;
   | %SVG.Description.class; %SVG.tref.extra.content; )*"
>
<!ELEMENT %SVG.tref.qname; %SVG.tref.content; >
<!-- end of SVG.tref.element -->]]>

<!ENTITY % SVG.tref.attlist "INCLUDE" >
<![%SVG.tref.attlist;[
<!ATTLIST %SVG.tref.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.TextContent.attrib;
  %SVG.Font.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
```

```
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.XLinkRequired.attrib;
  %SVG.External.attrib;
  x %Coordinates.datatype; #IMPLIED
  y %Coordinates.datatype; #IMPLIED
  dx %Lengths.datatype; #IMPLIED
  dy %Lengths.datatype; #IMPLIED
  rotate %Numbers.datatype; #IMPLIED
  textLength %Length.datatype; #IMPLIED
  lengthAdjust ( spacing | spacingAndGlyphs ) #IMPLIED
>
<!-- end of SVG.tref.attlist -->]]>

<!-- textPath: Text Path Element ..... -->

<!ENTITY % SVG.textPath.extra.content "" >

<!ENTITY % SVG.textPath.element "INCLUDE" >
<![%SVG.textPath.element;[
<!ENTITY % SVG.textPath.content
  "( #PCDATA | %SVG.tspan.qname; | %SVG.tref.qname; | %SVG.altGlyph.qname;
   | %SVG.animate.qname; | %SVG.set.qname; | %SVG.animateColor.qname;
   | %SVG.Description.class; %SVG.Hyperlink.class;
     %SVG.textPath.extra.content; )*"
>
<!ELEMENT %SVG.textPath.qname; %SVG.textPath.content; >
<!-- end of SVG.textPath.element -->]]>

<!ENTITY % SVG.textPath.attlist "INCLUDE" >
<![%SVG.textPath.attlist;[
<!ATTLIST %SVG.textPath.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.TextContent.attrib;
  %SVG.Font.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
```

```
  %SVG.Cursor.attrib;
  %SVG.XLinkRequired.attrib;
  %SVG.External.attrib;
  startOffset %Length.datatype; #IMPLIED
  textLength %Length.datatype; #IMPLIED
  lengthAdjust ( spacing | spacingAndGlyphs ) #IMPLIED
  method ( align | stretch ) #IMPLIED
  spacing ( auto | exact ) #IMPLIED
>
<!-- end of SVG.textPath.attlist -->]]>

<!-- altGlyph: Alternate Glyph Element ..... -->

<!ENTITY % SVG.altGlyph.extra.content "" >

<!ENTITY % SVG.altGlyph.element "INCLUDE" >
<![%SVG.altGlyph.element;[
<!ENTITY % SVG.altGlyph.content
  "( #PCDATA %SVG.altGlyph.extra.content; )*"
>
<!ELEMENT %SVG.altGlyph.qname; %SVG.altGlyph.content; >
<!-- end of SVG.altGlyph.element -->]]>

<!ENTITY % SVG.altGlyph.attlist "INCLUDE" >
<![%SVG.altGlyph.attlist;[
<!ATTLIST %SVG.altGlyph.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.TextContent.attrib;
  %SVG.Font.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.XLink.attrib;
  %SVG.External.attrib;
  x %Coordinates.datatype; #IMPLIED
  y %Coordinates.datatype; #IMPLIED
  dx %Lengths.datatype; #IMPLIED
  dy %Lengths.datatype; #IMPLIED
  glyphRef CDATA #IMPLIED
  format CDATA #IMPLIED
```

```
  rotate %Numbers.datatype; #IMPLIED
>
<!-- end of SVG.altGlyph.attlist -->]]>

<!-- altGlyphDef: Alternate Glyph Definition Element ... -->

<!ENTITY % SVG.altGlyphDef.extra.content "" >

<!ENTITY % SVG.altGlyphDef.element "INCLUDE" >
<![%SVG.altGlyphDef.element;[
<!ENTITY % SVG.altGlyphDef.content
  "(( %SVG.glyphRef.qname;+ | %SVG.altGlyphItem.qname;+ )
     %SVG.altGlyphDef.extra.content; )"
>
<!ELEMENT %SVG.altGlyphDef.qname; %SVG.altGlyphDef.content; >
<!-- end of SVG.altGlyphDef.element -->]]>

<!ENTITY % SVG.altGlyphDef.attlist "INCLUDE" >
<![%SVG.altGlyphDef.attlist;[
<!ATTLIST %SVG.altGlyphDef.qname;
  %SVG.Core.attrib;
>
<!-- end of SVG.altGlyphDef.attlist -->]]>

<!-- altGlyphItem: Alternate Glyph Item Element ..... -->

<!ENTITY % SVG.altGlyphItem.extra.content "" >

<!ENTITY % SVG.altGlyphItem.element "INCLUDE" >
<![%SVG.altGlyphItem.element;[
<!ENTITY % SVG.altGlyphItem.content
  "( %SVG.glyphRef.qname;+ %SVG.altGlyphItem.extra.content; )"
>
<!ELEMENT %SVG.altGlyphItem.qname; %SVG.altGlyphItem.content; >
<!-- end of SVG.altGlyphItem.element -->]]>

<!ENTITY % SVG.altGlyphItem.attlist "INCLUDE" >
<![%SVG.altGlyphItem.attlist;[
<!ATTLIST %SVG.altGlyphItem.qname;
  %SVG.Core.attrib;
>
<!-- end of SVG.altGlyphItem.attlist -->]]>

<!-- glyphRef: Glyph Reference Element ..... -->

<!ENTITY % SVG.glyphRef.element "INCLUDE" >
<![%SVG.glyphRef.element;[
<!ENTITY % SVG.glyphRef.content "EMPTY" >
```

```
<!ELEMENT %SVG.glyphRef.qname; %SVG.glyphRef.content; >
<!-- end of SVG.glyphRef.element -->]]>

<!ENTITY % SVG.glyphRef.attlist "INCLUDE" >
<![%SVG.glyphRef.attlist;[
<!ATTLIST %SVG.glyphRef.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Font.attrib;
  %SVG.XLink.attrib;
  x %Number.datatype; #IMPLIED
  y %Number.datatype; #IMPLIED
  dx %Number.datatype; #IMPLIED
  dy %Number.datatype; #IMPLIED
  glyphRef CDATA #IMPLIED
  format CDATA #IMPLIED
>
<!-- end of SVG.glyphRef.attlist -->]]>

<!-- end of svg-text.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-text.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Modular Framework Module  ..................................... -->
<!-- file: xhtml-framework-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-framework-1.mod,v 4.0 2001/04/02 22:42:49 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML Modular Framework 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-framework-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Modular Framework

   This required module instantiates the modules needed
   to support the XHTML modularization model, including:
```

```
      + datatypes
      + namespace-qualified names
      + common attributes
      + document model
      + character entities

   The Intrinsic Events module is ignored by default but
   occurs in this module because it must be instantiated
   prior to Attributes but after Datatypes.
-->

<!ENTITY % xhtml-arch.module "IGNORE" >
<![%xhtml-arch.module;[
<!ENTITY % xhtml-arch.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Base Architecture 1.0//EN"
         "xhtml-arch-1.mod" >
%xhtml-arch.mod;]]>

<!ENTITY % xhtml-notations.module "IGNORE" >
<![%xhtml-notations.module;[
<!ENTITY % xhtml-notations.mod
   PUBLIC "-//W3C//NOTATIONS XHTML Notations 1.0//EN"
         "xhtml-notations-1.mod" >
%xhtml-notations.mod;]]>

<!ENTITY % xhtml-datatypes.module "INCLUDE" >
<![%xhtml-datatypes.module;[
<!ENTITY % xhtml-datatypes.mod
   PUBLIC "-//W3C//ENTITIES XHTML Datatypes 1.0//EN"
         "xhtml-datatypes-1.mod" >
%xhtml-datatypes.mod;]]>

<!-- placeholder for XLink support module -->
<!ENTITY % xhtml-xlink.mod "" >
%xhtml-xlink.mod;

<!ENTITY % xhtml-qname.module "INCLUDE" >
<![%xhtml-qname.module;[
<!ENTITY % xhtml-qname.mod
   PUBLIC "-//W3C//ENTITIES XHTML Qualified Names 1.0//EN"
         "xhtml-qname-1.mod" >
%xhtml-qname.mod;]]>

<!ENTITY % xhtml-events.module "IGNORE" >
<![%xhtml-events.module;[
<!ENTITY % xhtml-events.mod
   PUBLIC "-//W3C//ENTITIES XHTML Intrinsic Events 1.0//EN"
         "xhtml-events-1.mod" >
```

```
%xhtml-events.mod;]]>

<!ENTITY % xhtml-attribs.module "INCLUDE" >
<![%xhtml-attribs.module;[
<!ENTITY % xhtml-attribs.mod
   PUBLIC "-//W3C//ENTITIES XHTML Common Attributes 1.0//EN"
         "xhtml-attribs-1.mod" >
%xhtml-attribs.mod;]]>

<!-- placeholder for content model redeclarations -->
<!ENTITY % xhtml-model.redecl "" >
%xhtml-model.redecl;

<!ENTITY % xhtml-model.module "INCLUDE" >
<![%xhtml-model.module;[
<!-- instantiate the Document Model module declared in the DTD driver
-->
%xhtml-model.mod;]]>

<!ENTITY % xhtml-charent.module "INCLUDE" >
<![%xhtml-charent.module;[
<!ENTITY % xhtml-charent.mod
   PUBLIC "-//W3C//ENTITIES XHTML Character Entities 1.0//EN"
         "xhtml-charent-1.mod" >
%xhtml-charent.mod;]]>

<!-- end of xhtml-framework-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-framework-1.mod
No license file was found, but licenses were detected in source scan.

<!--
   File isocyr1.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.
```

Entity names in this file are derived from files carrying the
following notice:

(C) International Organization for Standardization 1986
Permission to copy in any form is granted for use with
conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Russian Cyrillic//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isocyr1.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isocyr1 PUBLIC
"-//W3C//ENTITIES Russian Cyrillic//EN//XML"
"http://www.w3.org/2003/entities/2007/isocyr1.ent"
>
%isocyr1;

-->

<!ENTITY acy "а" ><!--CYRILLIC SMALL LETTER A -->
<!ENTITY Acy "А" ><!--CYRILLIC CAPITAL LETTER A -->
<!ENTITY bcy "б" ><!--CYRILLIC SMALL LETTER BE -->
<!ENTITY Bcy "Б" ><!--CYRILLIC CAPITAL LETTER BE -->
<!ENTITY chcy "ч" ><!--CYRILLIC SMALL LETTER CHE -->
<!ENTITY CHcy "Ч" ><!--CYRILLIC CAPITAL LETTER CHE -->
<!ENTITY dcy "д" ><!--CYRILLIC SMALL LETTER DE -->
<!ENTITY Dcy "Д" ><!--CYRILLIC CAPITAL LETTER DE -->
<!ENTITY ecy "э" ><!--CYRILLIC SMALL LETTER E -->
<!ENTITY Ecy "Э" ><!--CYRILLIC CAPITAL LETTER E -->
<!ENTITY fcy "ф" ><!--CYRILLIC SMALL LETTER EF -->
<!ENTITY Fcy "Ф" ><!--CYRILLIC CAPITAL LETTER EF -->
<!ENTITY gcy "г" ><!--CYRILLIC SMALL LETTER GHE -->
<!ENTITY Gcy "Г" ><!--CYRILLIC CAPITAL LETTER GHE -->
<!ENTITY hardcy "ъ" ><!--CYRILLIC SMALL LETTER HARD SIGN -->
<!ENTITY HARDcy "Ъ" ><!--CYRILLIC CAPITAL LETTER HARD SIGN -->
<!ENTITY icy "и" ><!--CYRILLIC SMALL LETTER I -->
<!ENTITY Icy "И" ><!--CYRILLIC CAPITAL LETTER I -->

```
<!ENTITY iecy          "&#x00435;" ><!--CYRILLIC SMALL LETTER IE -->
<!ENTITY IEcy          "&#x00415;" ><!--CYRILLIC CAPITAL LETTER IE -->
<!ENTITY iocy          "&#x00451;" ><!--CYRILLIC SMALL LETTER IO -->
<!ENTITY IOcy          "&#x00401;" ><!--CYRILLIC CAPITAL LETTER IO -->
<!ENTITY jcy           "&#x00439;" ><!--CYRILLIC SMALL LETTER SHORT I -->
<!ENTITY Jcy           "&#x00419;" ><!--CYRILLIC CAPITAL LETTER SHORT I -->
<!ENTITY kcy           "&#x0043A;" ><!--CYRILLIC SMALL LETTER KA -->
<!ENTITY Kcy           "&#x0041A;" ><!--CYRILLIC CAPITAL LETTER KA -->
<!ENTITY khcy          "&#x00445;" ><!--CYRILLIC SMALL LETTER HA -->
<!ENTITY KHcy          "&#x00425;" ><!--CYRILLIC CAPITAL LETTER HA -->
<!ENTITY lcy           "&#x0043B;" ><!--CYRILLIC SMALL LETTER EL -->
<!ENTITY Lcy           "&#x0041B;" ><!--CYRILLIC CAPITAL LETTER EL -->
<!ENTITY mcy           "&#x0043C;" ><!--CYRILLIC SMALL LETTER EM -->
<!ENTITY Mcy           "&#x0041C;" ><!--CYRILLIC CAPITAL LETTER EM -->
<!ENTITY ncy           "&#x0043D;" ><!--CYRILLIC SMALL LETTER EN -->
<!ENTITY Ncy           "&#x0041D;" ><!--CYRILLIC CAPITAL LETTER EN -->
<!ENTITY numero        "&#x02116;" ><!--NUMERO SIGN -->
<!ENTITY ocy           "&#x0043E;" ><!--CYRILLIC SMALL LETTER O -->
<!ENTITY Ocy           "&#x0041E;" ><!--CYRILLIC CAPITAL LETTER O -->
<!ENTITY pcy           "&#x0043F;" ><!--CYRILLIC SMALL LETTER PE -->
<!ENTITY Pcy           "&#x0041F;" ><!--CYRILLIC CAPITAL LETTER PE -->
<!ENTITY rcy           "&#x00440;" ><!--CYRILLIC SMALL LETTER ER -->
<!ENTITY Rcy           "&#x00420;" ><!--CYRILLIC CAPITAL LETTER ER -->
<!ENTITY scy           "&#x00441;" ><!--CYRILLIC SMALL LETTER ES -->
<!ENTITY Scy           "&#x00421;" ><!--CYRILLIC CAPITAL LETTER ES -->
<!ENTITY shchcy        "&#x00449;" ><!--CYRILLIC SMALL LETTER SHCHA -->
<!ENTITY SHCHcy        "&#x00429;" ><!--CYRILLIC CAPITAL LETTER SHCHA -->
<!ENTITY shcy          "&#x00448;" ><!--CYRILLIC SMALL LETTER SHA -->
<!ENTITY SHcy          "&#x00428;" ><!--CYRILLIC CAPITAL LETTER SHA -->
<!ENTITY softcy        "&#x0044C;" ><!--CYRILLIC SMALL LETTER SOFT SIGN -->
<!ENTITY SOFTcy        "&#x0042C;" ><!--CYRILLIC CAPITAL LETTER SOFT SIGN -->
<!ENTITY tcy           "&#x00442;" ><!--CYRILLIC SMALL LETTER TE -->
<!ENTITY Tcy           "&#x00422;" ><!--CYRILLIC CAPITAL LETTER TE -->
<!ENTITY tscy          "&#x00446;" ><!--CYRILLIC SMALL LETTER TSE -->
<!ENTITY TScy          "&#x00426;" ><!--CYRILLIC CAPITAL LETTER TSE -->
<!ENTITY ucy           "&#x00443;" ><!--CYRILLIC SMALL LETTER U -->
<!ENTITY Ucy           "&#x00423;" ><!--CYRILLIC CAPITAL LETTER U -->
<!ENTITY vcy           "&#x00432;" ><!--CYRILLIC SMALL LETTER VE -->
<!ENTITY Vcy           "&#x00412;" ><!--CYRILLIC CAPITAL LETTER VE -->
<!ENTITY yacy          "&#x0044F;" ><!--CYRILLIC SMALL LETTER YA -->
<!ENTITY YAcy          "&#x0042F;" ><!--CYRILLIC CAPITAL LETTER YA -->
<!ENTITY ycy           "&#x0044B;" ><!--CYRILLIC SMALL LETTER YERU -->
<!ENTITY Ycy           "&#x0042B;" ><!--CYRILLIC CAPITAL LETTER YERU -->
<!ENTITY yucy          "&#x0044E;" ><!--CYRILLIC SMALL LETTER YU -->
<!ENTITY YUcy          "&#x0042E;" ><!--CYRILLIC CAPITAL LETTER YU -->
<!ENTITY zcy           "&#x00437;" ><!--CYRILLIC SMALL LETTER ZE -->
<!ENTITY Zcy           "&#x00417;" ><!--CYRILLIC CAPITAL LETTER ZE -->
<!ENTITY zhcy          "&#x00436;" ><!--CYRILLIC SMALL LETTER ZHE -->
```

<!ENTITY ZHcy "Ж" ><!--CYRILLIC CAPITAL LETTER ZHE -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isocyr1.ent
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Tiny Document Model Module ..... -->
<!-- file: svg11-tiny-model.mod

This is SVG Tiny, a proper subset of SVG.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg11-tiny-model.mod,v 1.5 2002/11/14 15:11:03 fujisawa Exp $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Tiny Document Model//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11-tiny-model.mod"

..... -->

<!-- SVG 1.1 Tiny Document Model

This module describes the groupings of elements that make up
common content models for SVG elements.
-->

<!-- Content Models (Default) ..... -->

<!ENTITY % SVG.Style.class "" >
<!ENTITY % SVG.TextContent.class "" >
<!ENTITY % SVG.Marker.class "" >
<!ENTITY % SVG.ColorProfile.class "" >
<!ENTITY % SVG.Gradient.class "" >
<!ENTITY % SVG.Pattern.class "" >
<!ENTITY % SVG.Clip.class "" >
<!ENTITY % SVG.Mask.class "" >
<!ENTITY % SVG.Filter.class "" >
<!ENTITY % SVG.FilterPrimitive.class "" >
<!ENTITY % SVG.Cursor.class "" >
<!ENTITY % SVG.View.class "" >
<!ENTITY % SVG.Script.class "" >

<!-- module: svg-basic-structure.mod ..... -->

<!ENTITY % SVG.Description.extra.class "" >
<!ENTITY % SVG.Description.class

```
  "%SVG.desc.qname; | %SVG.title.qname; | %SVG.metadata.qname;
   %SVG.Description.extra.class;"
>

<!ENTITY % SVG.Use.extra.class "" >
<!ENTITY % SVG.Use.class
  "| %SVG.use.qname; %SVG.Use.extra.class;"
>

<!ENTITY % SVG.Structure.extra.class "" >
<!ENTITY % SVG.Structure.class
  "| %SVG.g.qname; | %SVG.defs.qname; %SVG.Use.class;
    %SVG.Structure.extra.class;"
>

<!-- module: svg-conditional.mod ..... -->

<!ENTITY % SVG.Conditional.extra.class "" >
<!ENTITY % SVG.Conditional.class
  "| %SVG.switch.qname; %SVG.Conditional.extra.class;"
>

<!-- module: svg-image.mod ..... -->

<!ENTITY % SVG.Image.extra.class "" >
<!ENTITY % SVG.Image.class
  "| %SVG.image.qname; %SVG.Image.extra.class;"
>

<!-- module: svg-shape.mod ..... -->

<!ENTITY % SVG.Shape.extra.class "" >
<!ENTITY % SVG.Shape.class
  "| %SVG.path.qname; | %SVG.rect.qname; | %SVG.circle.qname;
   | %SVG.line.qname; | %SVG.ellipse.qname; | %SVG.polyline.qname;
   | %SVG.polygon.qname; %SVG.Shape.extra.class;"
>

<!-- module: svg-basic-text.mod ..... -->

<!ENTITY % SVG.Text.extra.class "" >
<!ENTITY % SVG.Text.class
  "| %SVG.text.qname; %SVG.Text.extra.class;"
>

<!-- module: svg-hyperlink.mod ..... -->

<!ENTITY % SVG.Hyperlink.extra.class "" >
```

```
<!ENTITY % SVG.Hyperlink.class
  "| %SVG.a.qname; %SVG.Hyperlink.extra.class;"
>

<!-- module: svg-animation.mod ..... -->

<!ENTITY % SVG.Animation.extra.class "" >
<!ENTITY % SVG.Animation.class
  "%SVG.animate.qname; | %SVG.set.qname; | %SVG.animateMotion.qname; |
   %SVG.animateColor.qname; | %SVG.animateTransform.qname;
   %SVG.Animation.extra.class;"
>

<!-- module: svg-basic-font.mod ..... -->

<!ENTITY % SVG.Font.extra.class "" >
<!ENTITY % SVG.Font.class
  "| %SVG.font.qname; | %SVG.font-face.qname; %SVG.Font.extra.class;"
>

<!-- module: svg-extensibility.mod ..... -->

<!ENTITY % SVG.Extensibility.extra.class "" >
<!ENTITY % SVG.Extensibility.class
  "| %SVG.foreignObject.qname; %SVG.Extensibility.extra.class;"
>

<!-- end of svg11-tiny-model.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg11-tiny-model.mod
No license file was found, but licenses were detected in source scan.

<!--
   File isocyr2.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
```

(should) match the Unicode assignments in Unicode 5.x.

Entity names in this file are derived from files carrying the
following notice:

(C) International Organization for Standardization 1986
Permission to copy in any form is granted for use with
conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Non-Russian Cyrillic//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isocyr2.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isocyr2 PUBLIC
"-//W3C//ENTITIES Non-Russian Cyrillic//EN//XML"
"http://www.w3.org/2003/entities/2007/isocyr2.ent"
>
%isocyr2;

-->

<!ENTITY djcy "ђ" ><!--CYRILLIC SMALL LETTER DJE -->
<!ENTITY DJcy "Ђ" ><!--CYRILLIC CAPITAL LETTER DJE -->
<!ENTITY dscy "ѕ" ><!--CYRILLIC SMALL LETTER DZE -->
<!ENTITY DScy "Ѕ" ><!--CYRILLIC CAPITAL LETTER DZE -->
<!ENTITY dzcy "џ" ><!--CYRILLIC SMALL LETTER DZHE -->
<!ENTITY DZcy "Џ" ><!--CYRILLIC CAPITAL LETTER DZHE -->
<!ENTITY gjcy "ѓ" ><!--CYRILLIC SMALL LETTER GJE -->
<!ENTITY GJcy "Ѓ" ><!--CYRILLIC CAPITAL LETTER GJE -->
<!ENTITY iukcy "і" ><!--CYRILLIC SMALL LETTER BYELORUSSIAN-UKRAINIAN I -->
<!ENTITY Iukcy "І" ><!--CYRILLIC CAPITAL LETTER BYELORUSSIAN-UKRAINIAN I -->
<!ENTITY jsercy "ј" ><!--CYRILLIC SMALL LETTER JE -->
<!ENTITY Jsercy "Ј" ><!--CYRILLIC CAPITAL LETTER JE -->
<!ENTITY jukcy "є" ><!--CYRILLIC SMALL LETTER UKRAINIAN IE -->
<!ENTITY Jukcy "Є" ><!--CYRILLIC CAPITAL LETTER UKRAINIAN IE -->
<!ENTITY kjcy "ќ" ><!--CYRILLIC SMALL LETTER KJE -->
<!ENTITY KJcy "Ќ" ><!--CYRILLIC CAPITAL LETTER KJE -->
<!ENTITY ljcy "љ" ><!--CYRILLIC SMALL LETTER LJE -->

```
<!ENTITY LJcy           "&#x00409;" ><!--CYRILLIC CAPITAL LETTER LJE -->
<!ENTITY njcy           "&#x0045A;" ><!--CYRILLIC SMALL LETTER NJE -->
<!ENTITY NJcy           "&#x0040A;" ><!--CYRILLIC CAPITAL LETTER NJE -->
<!ENTITY tshcy          "&#x0045B;" ><!--CYRILLIC SMALL LETTER TSHE -->
<!ENTITY TSHcy          "&#x0040B;" ><!--CYRILLIC CAPITAL LETTER TSHE -->
<!ENTITY ubrcy          "&#x0045E;" ><!--CYRILLIC SMALL LETTER SHORT U -->
<!ENTITY Ubrcy          "&#x0040E;" ><!--CYRILLIC CAPITAL LETTER SHORT U -->
<!ENTITY yicy           "&#x00457;" ><!--CYRILLIC SMALL LETTER YI -->
<!ENTITY YIcy           "&#x00407;" ><!--CYRILLIC CAPITAL LETTER YI -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/mathml/mathml3/isocyr2.ent
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- SVG 1.1 Graphical Element Events Attribute Module ..... -->
<!-- file: svg-graphevents-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-graphevents-attrib.mod,v 1.2 2002/04/20 18:07:43 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Graphical Element Events Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-graphevents-attrib.mod"

   ..................................................................... -->

<!-- Graphical Element Events Attribute

     onfocusin, onfocusout, onactivate, onclick, onmousedown, onmouseup,
     onmouseover, onmousemove, onmouseout, onload

   This module defines the GraphicalEvents attribute set.
-->

<!ENTITY % SVG.onfocusin.attrib
  "onfocusin %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onfocusout.attrib
  "onfocusout %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onactivate.attrib
  "onactivate %Script.datatype; #IMPLIED"
```

```
>

<!ENTITY % SVG.onclick.attrib
  "onclick %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onmousedown.attrib
  "onmousedown %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onmouseup.attrib
  "onmouseup %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onmouseover.attrib
  "onmouseover %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onmousemove.attrib
  "onmousemove %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onmouseout.attrib
  "onmouseout %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onload.attrib
  "onload %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.GraphicalEvents.extra.attrib "" >

<!ENTITY % SVG.GraphicalEvents.attrib
  "%SVG.onfocusin.attrib;
   %SVG.onfocusout.attrib;
   %SVG.onactivate.attrib;
   %SVG.onclick.attrib;
   %SVG.onmousedown.attrib;
   %SVG.onmouseup.attrib;
   %SVG.onmouseover.attrib;
   %SVG.onmousemove.attrib;
   %SVG.onmouseout.attrib;
   %SVG.onload.attrib;
   %SVG.GraphicalEvents.extra.attrib;"
>

<!-- end of svg-graphevents-attrib.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-graphevents-attrib.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Inline Phrasal Module  ..................................... -->
<!-- file: xhtml-inlphras-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-inlphras-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Inline Phrasal 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-inlphras-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Inline Phrasal

      abbr, acronym, cite, code, dfn, em, kbd, q, samp, strong, var

   This module declares the elements and their attributes used to
   support inline-level phrasal markup.
-->

<!ENTITY % abbr.element  "INCLUDE" >
<![%abbr.element;[
<!ENTITY % abbr.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % abbr.qname  "abbr" >
<!ELEMENT %abbr.qname;  %abbr.content; >
<!-- end of abbr.element -->]]>

<!ENTITY % abbr.attlist  "INCLUDE" >
<![%abbr.attlist;[
<!ATTLIST %abbr.qname;
    %Common.attrib;
>
<!-- end of abbr.attlist -->]]>

<!ENTITY % acronym.element  "INCLUDE" >
<![%acronym.element;[
```

```
<!ENTITY % acronym.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % acronym.qname  "acronym" >
<!ELEMENT %acronym.qname;  %acronym.content; >
<!-- end of acronym.element -->]]>

<!ENTITY % acronym.attlist  "INCLUDE" >
<![%acronym.attlist;[
<!ATTLIST %acronym.qname;
    %Common.attrib;
>
<!-- end of acronym.attlist -->]]>

<!ENTITY % cite.element  "INCLUDE" >
<![%cite.element;[
<!ENTITY % cite.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % cite.qname  "cite" >
<!ELEMENT %cite.qname;  %cite.content; >
<!-- end of cite.element -->]]>

<!ENTITY % cite.attlist  "INCLUDE" >
<![%cite.attlist;[
<!ATTLIST %cite.qname;
    %Common.attrib;
>
<!-- end of cite.attlist -->]]>

<!ENTITY % code.element  "INCLUDE" >
<![%code.element;[
<!ENTITY % code.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % code.qname  "code" >
<!ELEMENT %code.qname;  %code.content; >
<!-- end of code.element -->]]>

<!ENTITY % code.attlist  "INCLUDE" >
<![%code.attlist;[
<!ATTLIST %code.qname;
    %Common.attrib;
>
<!-- end of code.attlist -->]]>

<!ENTITY % dfn.element  "INCLUDE" >
<![%dfn.element;[
```

```
<!ENTITY % dfn.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % dfn.qname  "dfn" >
<!ELEMENT %dfn.qname;  %dfn.content; >
<!-- end of dfn.element -->]]>

<!ENTITY % dfn.attlist  "INCLUDE" >
<![%dfn.attlist;[
<!ATTLIST %dfn.qname;
    %Common.attrib;
>
<!-- end of dfn.attlist -->]]>

<!ENTITY % em.element  "INCLUDE" >
<![%em.element;[
<!ENTITY % em.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % em.qname  "em" >
<!ELEMENT %em.qname;  %em.content; >
<!-- end of em.element -->]]>

<!ENTITY % em.attlist  "INCLUDE" >
<![%em.attlist;[
<!ATTLIST %em.qname;
    %Common.attrib;
>
<!-- end of em.attlist -->]]>

<!ENTITY % kbd.element  "INCLUDE" >
<![%kbd.element;[
<!ENTITY % kbd.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % kbd.qname  "kbd" >
<!ELEMENT %kbd.qname;  %kbd.content; >
<!-- end of kbd.element -->]]>

<!ENTITY % kbd.attlist  "INCLUDE" >
<![%kbd.attlist;[
<!ATTLIST %kbd.qname;
    %Common.attrib;
>
<!-- end of kbd.attlist -->]]>

<!ENTITY % q.element  "INCLUDE" >
<![%q.element;[
```

```
<!ENTITY % q.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % q.qname  "q" >
<!ELEMENT %q.qname;  %q.content; >
<!-- end of q.element -->]]>

<!ENTITY % q.attlist  "INCLUDE" >
<![%q.attlist;[
<!ATTLIST %q.qname;
    %Common.attrib;
    cite        %URI.datatype;          #IMPLIED
>
<!-- end of q.attlist -->]]>

<!ENTITY % samp.element  "INCLUDE" >
<![%samp.element;[
<!ENTITY % samp.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % samp.qname  "samp" >
<!ELEMENT %samp.qname;  %samp.content; >
<!-- end of samp.element -->]]>

<!ENTITY % samp.attlist  "INCLUDE" >
<![%samp.attlist;[
<!ATTLIST %samp.qname;
    %Common.attrib;
>
<!-- end of samp.attlist -->]]>

<!ENTITY % strong.element  "INCLUDE" >
<![%strong.element;[
<!ENTITY % strong.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % strong.qname  "strong" >
<!ELEMENT %strong.qname;  %strong.content; >
<!-- end of strong.element -->]]>

<!ENTITY % strong.attlist  "INCLUDE" >
<![%strong.attlist;[
<!ATTLIST %strong.qname;
    %Common.attrib;
>
<!-- end of strong.attlist -->]]>

<!ENTITY % var.element  "INCLUDE" >
```

```
<![%var.element;[
<!ENTITY % var.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % var.qname  "var" >
<!ELEMENT %var.qname;  %var.content; >
<!-- end of var.element -->]]>

<!ENTITY % var.attlist  "INCLUDE" >
<![%var.attlist;[
<!ATTLIST %var.qname;
    %Common.attrib;
>
<!-- end of var.attlist -->]]>

<!-- end of xhtml-inlphras-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-inlphras-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- SVG 1.1 Document Events Attribute Module ............................. -->
<!-- file: svg-docevents-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-docevents-attrib.mod,v 1.2 2002/04/20 18:07:42 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Document Events Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-docevents-attrib.mod"

   ....................................................................... -->

<!-- Document Events Attribute

     onunload, onabort, onerror, onresize, onscroll, onzoom

   This module defines the DocumentEvents attribute set.
-->

<!ENTITY % SVG.onunload.attrib
  "onunload %Script.datatype; #IMPLIED"
>
```

```
<!ENTITY % SVG.onabort.attrib
  "onabort %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onerror.attrib
  "onerror %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onresize.attrib
  "onresize %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onscroll.attrib
  "onscroll %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onzoom.attrib
  "onzoom %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.DocumentEvents.extra.attrib "" >

<!ENTITY % SVG.DocumentEvents.attrib
  "%SVG.onunload.attrib;
   %SVG.onabort.attrib;
   %SVG.onerror.attrib;
   %SVG.onresize.attrib;
   %SVG.onscroll.attrib;
   %SVG.onzoom.attrib;
   %SVG.DocumentEvents.extra.attrib;"
>

<!-- end of svg-docevents-attrib.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-docevents-attrib.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Structure Module ............................................ -->
<!-- file: svg-structure.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-structure.mod,v 1.5 2002/11/14 15:11:03 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:
```

```
      PUBLIC "-//W3C//ELEMENTS SVG 1.1 Structure//EN"
      SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-structure.mod"

    ..................................................................... -->

<!-- Structure

      svg, g, defs, desc, title, metadata, symbol, use

    This module declares the major structural elements and their attributes.
-->

<!-- Qualified Names (Default) ....................... -->

<!ENTITY % SVG.svg.qname "svg" >
<!ENTITY % SVG.g.qname "g" >
<!ENTITY % SVG.defs.qname "defs" >
<!ENTITY % SVG.desc.qname "desc" >
<!ENTITY % SVG.title.qname "title" >
<!ENTITY % SVG.metadata.qname "metadata" >
<!ENTITY % SVG.symbol.qname "symbol" >
<!ENTITY % SVG.use.qname "use" >

<!-- Attribute Collections (Default) ..................... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.DocumentEvents.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
```

```
<!ENTITY % SVG.XLinkEmbed.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Description.class ..... -->

<!ENTITY % SVG.Description.extra.class "" >

<!ENTITY % SVG.Description.class
  "%SVG.desc.qname; | %SVG.title.qname; | %SVG.metadata.qname;
   %SVG.Description.extra.class;"
>

<!-- SVG.Use.class ..... -->

<!ENTITY % SVG.Use.extra.class "" >

<!ENTITY % SVG.Use.class
  "| %SVG.use.qname; %SVG.Use.extra.class;"
>

<!-- SVG.Structure.class ..... -->

<!ENTITY % SVG.Structure.extra.class "" >

<!ENTITY % SVG.Structure.class
  "| %SVG.svg.qname; | %SVG.g.qname; | %SVG.defs.qname; | %SVG.symbol.qname;
     %SVG.Use.class; %SVG.Structure.extra.class;"
>

<!-- SVG.Presentation.attrib ..... -->

<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
   %SVG.Viewport.attrib;
   %SVG.Text.attrib;
   %SVG.TextContent.attrib;
   %SVG.Font.attrib;
   %SVG.Paint.attrib;
   %SVG.Color.attrib;
   %SVG.Opacity.attrib;
   %SVG.Graphics.attrib;
   %SVG.Marker.attrib;
   %SVG.ColorProfile.attrib;
   %SVG.Gradient.attrib;
   %SVG.Clip.attrib;
   %SVG.Mask.attrib;
```

```
    %SVG.Filter.attrib;
    %SVG.FilterColor.attrib;
    %SVG.Cursor.attrib;
    flood-color %SVGColor.datatype; #IMPLIED
    flood-opacity %OpacityValue.datatype; #IMPLIED
    lighting-color %SVGColor.datatype; #IMPLIED
    %SVG.Presentation.extra.attrib;"
>

<!-- svg: SVG Document Element ..... -->

<!ENTITY % SVG.svg.extra.content "" >

<!ENTITY % SVG.svg.element "INCLUDE" >
<![%SVG.svg.element;[
<!ENTITY % SVG.svg.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
     %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
     %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
     %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
     %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
     %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
     %SVG.Font.class; %SVG.svg.extra.content; )*"
>
<!ELEMENT %SVG.svg.qname; %SVG.svg.content; >
<!-- end of SVG.svg.element -->]]>

<!ENTITY % SVG.svg.attlist "INCLUDE" >
<![%SVG.svg.attlist;[
<!ATTLIST %SVG.svg.qname;
  %SVG.xmlns.attrib;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.DocumentEvents.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #IMPLIED
  height %Length.datatype; #IMPLIED
  viewBox %ViewBoxSpec.datatype; #IMPLIED
  preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
  zoomAndPan ( disable | magnify ) 'magnify'
  version %Number.datatype; #FIXED '1.1'
  baseProfile %Text.datatype; #IMPLIED
  contentScriptType %ContentType.datatype; 'text/ecmascript'
```

```
  contentStyleType %ContentType.datatype; 'text/css'
>
<!-- end of SVG.svg.attlist -->]]>

<!-- g: Group Element ..... -->

<!ENTITY % SVG.g.extra.content "" >

<!ENTITY % SVG.g.element "INCLUDE" >
<![%SVG.g.element;[
<!ENTITY % SVG.g.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
    %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
    %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
    %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
    %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
    %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
    %SVG.Font.class; %SVG.g.extra.content; )*"
>
<!ELEMENT %SVG.g.qname; %SVG.g.content; >
<!-- end of SVG.g.element -->]]>

<!ENTITY % SVG.g.attlist "INCLUDE" >
<![%SVG.g.attlist;[
<!ATTLIST %SVG.g.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.External.attrib;
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.g.attlist -->]]>

<!-- defs: Definisions Element ..... -->

<!ENTITY % SVG.defs.extra.content "" >

<!ENTITY % SVG.defs.element "INCLUDE" >
<![%SVG.defs.element;[
<!ENTITY % SVG.defs.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
    %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
    %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
    %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
    %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
    %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
```

```
    %SVG.Font.class; %SVG.defs.extra.content; )*"
>
<!ELEMENT %SVG.defs.qname; %SVG.defs.content; >
<!-- end of SVG.defs.element -->]]>

<!ENTITY % SVG.defs.attlist "INCLUDE" >
<![%SVG.defs.attlist;[
<!ATTLIST %SVG.defs.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.External.attrib;
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.defs.attlist -->]]>

<!-- desc: Description Element ..... -->

<!ENTITY % SVG.desc.extra.content "" >

<!ENTITY % SVG.desc.element "INCLUDE" >
<![%SVG.desc.element;[
<!ENTITY % SVG.desc.content
  "( #PCDATA %SVG.desc.extra.content; )*"
>
<!ELEMENT %SVG.desc.qname; %SVG.desc.content; >
<!-- end of SVG.desc.element -->]]>

<!ENTITY % SVG.desc.attlist "INCLUDE" >
<![%SVG.desc.attlist;[
<!ATTLIST %SVG.desc.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
>
<!-- end of SVG.desc.attlist -->]]>

<!-- title: Title Element ..... -->

<!ENTITY % SVG.title.extra.content "" >

<!ENTITY % SVG.title.element "INCLUDE" >
<![%SVG.title.element;[
<!ENTITY % SVG.title.content
  "( #PCDATA %SVG.title.extra.content; )*"
>
<!ELEMENT %SVG.title.qname; %SVG.title.content; >
```

```
<!-- end of SVG.title.element -->]]>

<!ENTITY % SVG.title.attlist "INCLUDE" >
<![%SVG.title.attlist;[
<!ATTLIST %SVG.title.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
>
<!-- end of SVG.title.attlist -->]]>

<!-- metadata: Metadata Element ..... -->

<!ENTITY % SVG.metadata.extra.content "" >

<!ENTITY % SVG.metadata.element "INCLUDE" >
<![%SVG.metadata.element;[
<!ENTITY % SVG.metadata.content
  "( #PCDATA %SVG.metadata.extra.content; )*"
>
<!ELEMENT %SVG.metadata.qname; %SVG.metadata.content; >
<!-- end of SVG.metadata.element -->]]>

<!ENTITY % SVG.metadata.attlist "INCLUDE" >
<![%SVG.metadata.attlist;[
<!ATTLIST %SVG.metadata.qname;
  %SVG.Core.attrib;
>
<!-- end of SVG.metadata.attlist -->]]>

<!-- symbol: Symbol Element ..... -->

<!ENTITY % SVG.symbol.extra.content "" >

<!ENTITY % SVG.symbol.element "INCLUDE" >
<![%SVG.symbol.element;[
<!ENTITY % SVG.symbol.content
  "( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
    %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
    %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class;
    %SVG.ColorProfile.class; %SVG.Gradient.class; %SVG.Pattern.class;
    %SVG.Clip.class; %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
    %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
    %SVG.Font.class; %SVG.symbol.extra.content; )*"
>
<!ELEMENT %SVG.symbol.qname; %SVG.symbol.content; >
<!-- end of SVG.symbol.element -->]]>

<!ENTITY % SVG.symbol.attlist "INCLUDE" >
```

```
<![%SVG.symbol.attlist;[
<!ATTLIST %SVG.symbol.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.External.attrib;
  viewBox %ViewBoxSpec.datatype; #IMPLIED
  preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
>
<!-- end of SVG.symbol.attlist -->]]>

<!-- use: Use Element ..... -->

<!ENTITY % SVG.use.extra.content "" >

<!ENTITY % SVG.use.element "INCLUDE" >
<![%SVG.use.element;[
<!ENTITY % SVG.use.content
  "(( %SVG.Description.class; )*, ( %SVG.Animation.class;
     %SVG.use.extra.content; )*)"
>
<!ELEMENT %SVG.use.qname; %SVG.use.content; >
<!-- end of SVG.use.element -->]]>

<!ENTITY % SVG.use.attlist "INCLUDE" >
<![%SVG.use.attlist;[
<!ATTLIST %SVG.use.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.XLinkEmbed.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #IMPLIED
  height %Length.datatype; #IMPLIED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.use.attlist -->]]>

<!-- end of svg-structure.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-structure.mod

No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- SVG 1.1 Filter Module ..................................................... -->
<!-- file: svg-filter.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-filter.mod,v 1.4 2002/11/14 15:11:02 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Filter//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-filter.mod"

   ..................................................................... -->

<!-- Filter

     filter, feBlend, feColorMatrix, feComponentTransfer, feComposite,
     feConvolveMatrix, feDiffuseLighting, feDisplacementMap, feFlood,
     feGaussianBlur, feImage, feMerge, feMergeNode, feMorphology, feOffset,
     feSpecularLighting, feTile, feTurbulence, feDistantLight, fePointLight,
     feSpotLight, feFuncR, feFuncG, feFuncB, feFuncA

   This module declares markup to provide support for filter effect.
-->

<!-- 'filter' property/attribute value (e.g., 'none', <uri>) -->
<!ENTITY % FilterValue.datatype "CDATA" >

<!-- list of <number>s, but at least one and at most two -->
<!ENTITY % NumberOptionalNumber.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.filter.qname "filter" >
<!ENTITY % SVG.feBlend.qname "feBlend" >
<!ENTITY % SVG.feColorMatrix.qname "feColorMatrix" >
<!ENTITY % SVG.feComponentTransfer.qname "feComponentTransfer" >
<!ENTITY % SVG.feComposite.qname "feComposite" >
<!ENTITY % SVG.feConvolveMatrix.qname "feConvolveMatrix" >
<!ENTITY % SVG.feDiffuseLighting.qname "feDiffuseLighting" >
<!ENTITY % SVG.feDisplacementMap.qname "feDisplacementMap" >
<!ENTITY % SVG.feFlood.qname "feFlood" >
<!ENTITY % SVG.feGaussianBlur.qname "feGaussianBlur" >
<!ENTITY % SVG.feImage.qname "feImage" >
<!ENTITY % SVG.feMerge.qname "feMerge" >
```

```
<!ENTITY % SVG.feMergeNode.qname "feMergeNode" >
<!ENTITY % SVG.feMorphology.qname "feMorphology" >
<!ENTITY % SVG.feOffset.qname "feOffset" >
<!ENTITY % SVG.feSpecularLighting.qname "feSpecularLighting" >
<!ENTITY % SVG.feTile.qname "feTile" >
<!ENTITY % SVG.feTurbulence.qname "feTurbulence" >
<!ENTITY % SVG.feDistantLight.qname "feDistantLight" >
<!ENTITY % SVG.fePointLight.qname "fePointLight" >
<!ENTITY % SVG.feSpotLight.qname "feSpotLight" >
<!ENTITY % SVG.feFuncR.qname "feFuncR" >
<!ENTITY % SVG.feFuncG.qname "feFuncG" >
<!ENTITY % SVG.feFuncB.qname "feFuncB" >
<!ENTITY % SVG.feFuncA.qname "feFuncA" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.XLinkEmbed.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Filter.class ..... -->

<!ENTITY % SVG.Filter.extra.class "" >

<!ENTITY % SVG.Filter.class
  "| %SVG.filter.qname; %SVG.Filter.extra.class;"
>

<!-- SVG.FilterPrimitive.class ..... -->

<!ENTITY % SVG.FilterPrimitive.extra.class "" >
```

```
<!ENTITY % SVG.FilterPrimitive.class
   "| %SVG.feBlend.qname; | %SVG.feColorMatrix.qname;
    | %SVG.feComponentTransfer.qname; | %SVG.feComposite.qname;
    | %SVG.feConvolveMatrix.qname; | %SVG.feDiffuseLighting.qname;
    | %SVG.feDisplacementMap.qname; | %SVG.feFlood.qname;
    | %SVG.feGaussianBlur.qname; | %SVG.feImage.qname; | %SVG.feMerge.qname;
    | %SVG.feMorphology.qname; | %SVG.feOffset.qname;
    | %SVG.feSpecularLighting.qname; | %SVG.feTile.qname;
    | %SVG.feTurbulence.qname; %SVG.FilterPrimitive.extra.class;"
>

<!-- SVG.Filter.attrib ..... -->

<!ENTITY % SVG.Filter.extra.attrib "" >

<!ENTITY % SVG.Filter.attrib
   "filter %FilterValue.datatype; #IMPLIED
    %SVG.Filter.extra.attrib;"
>

<!-- SVG.FilterColor.attrib ..... -->

<!ENTITY % SVG.FilterColor.extra.attrib "" >

<!ENTITY % SVG.FilterColor.attrib
   "color-interpolation-filters ( auto | sRGB | linearRGB | inherit )
                                #IMPLIED
    %SVG.FilterColor.extra.attrib;"
>

<!-- SVG.FilterPrimitive.attrib ..... -->

<!ENTITY % SVG.FilterPrimitive.extra.attrib "" >

<!ENTITY % SVG.FilterPrimitive.attrib
   "x %Coordinate.datatype; #IMPLIED
    y %Coordinate.datatype; #IMPLIED
    width %Length.datatype; #IMPLIED
    height %Length.datatype; #IMPLIED
    result CDATA #IMPLIED
    %SVG.FilterPrimitive.extra.attrib;"
>

<!-- SVG.FilterPrimitiveWithIn.attrib ..... -->

<!ENTITY % SVG.FilterPrimitiveWithIn.extra.attrib "" >
```

```
<!ENTITY % SVG.FilterPrimitiveWithIn.attrib
  "%SVG.FilterPrimitive.attrib;
   in CDATA #IMPLIED
   %SVG.FilterPrimitiveWithIn.extra.attrib;"
>

<!-- SVG.Presentation.attrib ..... -->

<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
   %SVG.Viewport.attrib;
   %SVG.Text.attrib;
   %SVG.TextContent.attrib;
   %SVG.Font.attrib;
   %SVG.Paint.attrib;
   %SVG.Color.attrib;
   %SVG.Opacity.attrib;
   %SVG.Graphics.attrib;
   %SVG.Marker.attrib;
   %SVG.ColorProfile.attrib;
   %SVG.Gradient.attrib;
   %SVG.Clip.attrib;
   %SVG.Mask.attrib;
   %SVG.Filter.attrib;
   %SVG.FilterColor.attrib;
   %SVG.Cursor.attrib;
   flood-color %SVGColor.datatype; #IMPLIED
   flood-opacity %OpacityValue.datatype; #IMPLIED
   lighting-color %SVGColor.datatype; #IMPLIED
   %SVG.Presentation.extra.attrib;"
>

<!-- filter: Filter Element ..... -->

<!ENTITY % SVG.filter.extra.content "" >

<!ENTITY % SVG.filter.element "INCLUDE" >
<![%SVG.filter.element;[
<!ENTITY % SVG.filter.content
  "(( %SVG.Description.class; )*, ( %SVG.animate.qname; | %SVG.set.qname;
     %SVG.FilterPrimitive.class; %SVG.filter.extra.content; )*)"
>
<!ELEMENT %SVG.filter.qname; %SVG.filter.content; >
<!-- end of SVG.filter.element -->]]>

<!ENTITY % SVG.filter.attlist "INCLUDE" >
```

```
<![%SVG.filter.attlist;[
<!ATTLIST %SVG.filter.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.XLink.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #IMPLIED
  height %Length.datatype; #IMPLIED
  filterRes %NumberOptionalNumber.datatype; #IMPLIED
  filterUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
  primitiveUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
>
<!-- end of SVG.filter.attlist -->]]>

<!-- feBlend: Filter Effect Blend Element ..... -->

<!ENTITY % SVG.feBlend.extra.content "" >

<!ENTITY % SVG.feBlend.element "INCLUDE" >
<![%SVG.feBlend.element;[
<!ENTITY % SVG.feBlend.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feBlend.extra.content; )*"
>
<!ELEMENT %SVG.feBlend.qname; %SVG.feBlend.content; >
<!-- end of SVG.feBlend.element -->]]>

<!ENTITY % SVG.feBlend.attlist "INCLUDE" >
<![%SVG.feBlend.attlist;[
<!ATTLIST %SVG.feBlend.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  in2 CDATA #REQUIRED
  mode ( normal | multiply | screen | darken | lighten ) 'normal'
>
<!-- end of SVG.feBlend.attlist -->]]>

<!-- feColorMatrix: Filter Effect Color Matrix Element . -->

<!ENTITY % SVG.feColorMatrix.extra.content "" >

<!ENTITY % SVG.feColorMatrix.element "INCLUDE" >
<![%SVG.feColorMatrix.element;[
<!ENTITY % SVG.feColorMatrix.content
  "( %SVG.animate.qname; | %SVG.set.qname;
```

```
    %SVG.feColorMatrix.extra.content; )*"
>
<!ELEMENT %SVG.feColorMatrix.qname; %SVG.feColorMatrix.content; >
<!-- end of SVG.feColorMatrix.element -->]]>

<!ENTITY % SVG.feColorMatrix.attlist "INCLUDE" >
<![%SVG.feColorMatrix.attlist;[
<!ATTLIST %SVG.feColorMatrix.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  type ( matrix | saturate | hueRotate | luminanceToAlpha ) 'matrix'
  values CDATA #IMPLIED
>
<!-- end of SVG.feColorMatrix.attlist -->]]>

<!-- feComponentTransfer: Filter Effect Component Transfer Element -->

<!ENTITY % SVG.feComponentTransfer.extra.content "" >

<!ENTITY % SVG.feComponentTransfer.element "INCLUDE" >
<![%SVG.feComponentTransfer.element;[
<!ENTITY % SVG.feComponentTransfer.content
  "( %SVG.feFuncR.qname;?, %SVG.feFuncG.qname;?, %SVG.feFuncB.qname;?,
    %SVG.feFuncA.qname;? %SVG.feComponentTransfer.extra.content; )"
>
<!ELEMENT %SVG.feComponentTransfer.qname; %SVG.feComponentTransfer.content; >
<!-- end of SVG.feComponentTransfer.element -->]]>

<!ENTITY % SVG.feComponentTransfer.attlist "INCLUDE" >
<![%SVG.feComponentTransfer.attlist;[
<!ATTLIST %SVG.feComponentTransfer.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
>
<!-- end of SVG.feComponentTransfer.attlist -->]]>

<!-- feComposite: Filter Effect Composite Element ..... -->

<!ENTITY % SVG.feComposite.extra.content "" >

<!ENTITY % SVG.feComposite.element "INCLUDE" >
<![%SVG.feComposite.element;[
<!ENTITY % SVG.feComposite.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feComposite.extra.content; )*"
>
<!ELEMENT %SVG.feComposite.qname; %SVG.feComposite.content; >
```

```
<!-- end of SVG.feComposite.element -->]]>

<!ENTITY % SVG.feComposite.attlist "INCLUDE" >
<![%SVG.feComposite.attlist;[
<!ATTLIST %SVG.feComposite.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  in2 CDATA #REQUIRED
  operator ( over | in | out | atop | xor | arithmetic ) 'over'
  k1 %Number.datatype; #IMPLIED
  k2 %Number.datatype; #IMPLIED
  k3 %Number.datatype; #IMPLIED
  k4 %Number.datatype; #IMPLIED
>
<!-- end of SVG.feComposite.attlist -->]]>

<!-- feConvolveMatrix: Filter Effect Convolve Matrix Element -->

<!ENTITY % SVG.feConvolveMatrix.extra.content "" >

<!ENTITY % SVG.feConvolveMatrix.element "INCLUDE" >
<![%SVG.feConvolveMatrix.element;[
<!ENTITY % SVG.feConvolveMatrix.content
  "( %SVG.animate.qname; | %SVG.set.qname;
     %SVG.feConvolveMatrix.extra.content; )*"
>
<!ELEMENT %SVG.feConvolveMatrix.qname; %SVG.feConvolveMatrix.content; >
<!-- end of SVG.feConvolveMatrix.element -->]]>

<!ENTITY % SVG.feConvolveMatrix.attlist "INCLUDE" >
<![%SVG.feConvolveMatrix.attlist;[
<!ATTLIST %SVG.feConvolveMatrix.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  order %NumberOptionalNumber.datatype; #REQUIRED
  kernelMatrix CDATA #REQUIRED
  divisor %Number.datatype; #IMPLIED
  bias %Number.datatype; #IMPLIED
  targetX %Integer.datatype; #IMPLIED
  targetY %Integer.datatype; #IMPLIED
  edgeMode ( duplicate | wrap | none ) 'duplicate'
  kernelUnitLength %NumberOptionalNumber.datatype; #IMPLIED
  preserveAlpha %Boolean.datatype; #IMPLIED
>
<!-- end of SVG.feConvolveMatrix.attlist -->]]>
```

```
<!-- feDiffuseLighting: Filter Effect Diffuse Lighting Element -->

<!ENTITY % SVG.feDiffuseLighting.extra.content "" >

<!ENTITY % SVG.feDiffuseLighting.element "INCLUDE" >
<![%SVG.feDiffuseLighting.element;[
<!ENTITY % SVG.feDiffuseLighting.content
  "(( %SVG.feDistantLight.qname; | %SVG.fePointLight.qname;
    | %SVG.feSpotLight.qname; ), ( %SVG.animate.qname; | %SVG.set.qname;
    | %SVG.animateColor.qname; %SVG.feDiffuseLighting.extra.content; )*)"
>
<!ELEMENT %SVG.feDiffuseLighting.qname; %SVG.feDiffuseLighting.content; >
<!-- end of SVG.feDiffuseLighting.element -->]]>

<!ENTITY % SVG.feDiffuseLighting.attlist "INCLUDE" >
<![%SVG.feDiffuseLighting.attlist;[
<!ATTLIST %SVG.feDiffuseLighting.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Color.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  lighting-color %SVGColor.datatype; #IMPLIED
  surfaceScale %Number.datatype; #IMPLIED
  diffuseConstant %Number.datatype; #IMPLIED
  kernelUnitLength %NumberOptionalNumber.datatype; #IMPLIED
>
<!-- end of SVG.feDiffuseLighting.attlist -->]]>

<!-- feDisplacementMap: Filter Effect Displacement Map Element -->

<!ENTITY % SVG.feDisplacementMap.extra.content "" >

<!ENTITY % SVG.feDisplacementMap.element "INCLUDE" >
<![%SVG.feDisplacementMap.element;[
<!ENTITY % SVG.feDisplacementMap.content
  "( %SVG.animate.qname; | %SVG.set.qname;
     %SVG.feDisplacementMap.extra.content; )*"
>
<!ELEMENT %SVG.feDisplacementMap.qname; %SVG.feDisplacementMap.content; >
<!-- end of SVG.feDisplacementMap.element -->]]>

<!ENTITY % SVG.feDisplacementMap.attlist "INCLUDE" >
<![%SVG.feDisplacementMap.attlist;[
<!ATTLIST %SVG.feDisplacementMap.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
```

```
  in2 CDATA #REQUIRED
  scale %Number.datatype; #IMPLIED
  xChannelSelector ( R | G | B | A ) 'A'
  yChannelSelector ( R | G | B | A ) 'A'
>
<!-- end of SVG.feDisplacementMap.attlist -->]]>

<!-- feFlood: Filter Effect Flood Element ..... -->

<!ENTITY % SVG.feFlood.extra.content "" >

<!ENTITY % SVG.feFlood.element "INCLUDE" >
<![%SVG.feFlood.element;[
<!ENTITY % SVG.feFlood.content
  "( %SVG.animate.qname; | %SVG.set.qname; | %SVG.animateColor.qname;
    %SVG.feFlood.extra.content; )*"
>
<!ELEMENT %SVG.feFlood.qname; %SVG.feFlood.content; >
<!-- end of SVG.feFlood.element -->]]>

<!ENTITY % SVG.feFlood.attlist "INCLUDE" >
<![%SVG.feFlood.attlist;[
<!ATTLIST %SVG.feFlood.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Color.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  flood-color %SVGColor.datatype; #IMPLIED
  flood-opacity %OpacityValue.datatype; #IMPLIED
>
<!-- end of SVG.feFlood.attlist -->]]>

<!-- feGaussianBlur: Filter Effect Gaussian Blur Element -->

<!ENTITY % SVG.feGaussianBlur.extra.content "" >

<!ENTITY % SVG.feGaussianBlur.element "INCLUDE" >
<![%SVG.feGaussianBlur.element;[
<!ENTITY % SVG.feGaussianBlur.content
  "( %SVG.animate.qname; | %SVG.set.qname;
    %SVG.feGaussianBlur.extra.content; )*"
>
<!ELEMENT %SVG.feGaussianBlur.qname; %SVG.feGaussianBlur.content; >
<!-- end of SVG.feGaussianBlur.element -->]]>

<!ENTITY % SVG.feGaussianBlur.attlist "INCLUDE" >
<![%SVG.feGaussianBlur.attlist;[
```

```
<!ATTLIST %SVG.feGaussianBlur.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  stdDeviation %NumberOptionalNumber.datatype; #IMPLIED
>
<!-- end of SVG.feGaussianBlur.attlist -->]]>

<!-- feImage: Filter Effect Image Element ..... -->

<!ENTITY % SVG.feImage.extra.content "" >

<!ENTITY % SVG.feImage.element "INCLUDE" >
<![%SVG.feImage.element;[
<!ENTITY % SVG.feImage.content
  "( %SVG.animate.qname; | %SVG.set.qname; | %SVG.animateTransform.qname;
     %SVG.feImage.extra.content; )*"
>
<!ELEMENT %SVG.feImage.qname; %SVG.feImage.content; >
<!-- end of SVG.feImage.element -->]]>

<!ENTITY % SVG.feImage.attlist "INCLUDE" >
<![%SVG.feImage.attlist;[
<!ATTLIST %SVG.feImage.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.FilterPrimitive.attrib;
  %SVG.XLinkEmbed.attrib;
  %SVG.External.attrib;
  preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
>
<!-- end of SVG.feImage.attlist -->]]>

<!-- feMerge: Filter Effect Merge Element ..... -->

<!ENTITY % SVG.feMerge.extra.content "" >

<!ENTITY % SVG.feMerge.element "INCLUDE" >
<![%SVG.feMerge.element;[
<!ENTITY % SVG.feMerge.content
  "( %SVG.feMergeNode.qname; %SVG.feMerge.extra.content; )*"
>
<!ELEMENT %SVG.feMerge.qname; %SVG.feMerge.content; >
<!-- end of SVG.feMerge.element -->]]>

<!ENTITY % SVG.feMerge.attlist "INCLUDE" >
<![%SVG.feMerge.attlist;[
```

```
<!ATTLIST %SVG.feMerge.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitive.attrib;
>
<!-- end of SVG.feMerge.attlist -->]]>

<!-- feMergeNode: Filter Effect Merge Node Element ..... -->

<!ENTITY % SVG.feMergeNode.extra.content "" >

<!ENTITY % SVG.feMergeNode.element "INCLUDE" >
<![%SVG.feMergeNode.element;[
<!ENTITY % SVG.feMergeNode.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feMergeNode.extra.content; )*"
>
<!ELEMENT %SVG.feMergeNode.qname; %SVG.feMergeNode.content; >
<!-- end of SVG.feMergeNode.element -->]]>

<!ENTITY % SVG.feMergeNode.attlist "INCLUDE" >
<![%SVG.feMergeNode.attlist;[
<!ATTLIST %SVG.feMergeNode.qname;
  %SVG.Core.attrib;
  in CDATA #IMPLIED
>
<!-- end of SVG.feMergeNode.attlist -->]]>

<!-- feMorphology: Filter Effect Morphology Element .... -->

<!ENTITY % SVG.feMorphology.extra.content "" >

<!ENTITY % SVG.feMorphology.element "INCLUDE" >
<![%SVG.feMorphology.element;[
<!ENTITY % SVG.feMorphology.content
  "( %SVG.animate.qname; | %SVG.set.qname;
    %SVG.feMorphology.extra.content; )*"
>
<!ELEMENT %SVG.feMorphology.qname; %SVG.feMorphology.content; >
<!-- end of SVG.feMorphology.element -->]]>

<!ENTITY % SVG.feMorphology.attlist "INCLUDE" >
<![%SVG.feMorphology.attlist;[
<!ATTLIST %SVG.feMorphology.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  operator ( erode | dilate ) 'erode'
  radius %NumberOptionalNumber.datatype; #IMPLIED
```

```
>
<!-- end of SVG.feMorphology.attlist -->]]>

<!-- feOffset: Filter Effect Offset Element ..... -->

<!ENTITY % SVG.feOffset.extra.content "" >

<!ENTITY % SVG.feOffset.element "INCLUDE" >
<![%SVG.feOffset.element;[
<!ENTITY % SVG.feOffset.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feOffset.extra.content; )*"
>
<!ELEMENT %SVG.feOffset.qname; %SVG.feOffset.content; >
<!-- end of SVG.feOffset.element -->]]>

<!ENTITY % SVG.feOffset.attlist "INCLUDE" >
<![%SVG.feOffset.attlist;[
<!ATTLIST %SVG.feOffset.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  dx %Number.datatype; #IMPLIED
  dy %Number.datatype; #IMPLIED
>
<!-- end of SVG.feOffset.attlist -->]]>

<!-- feSpecularLighting: Filter Effect Specular Lighting Element -->

<!ENTITY % SVG.feSpecularLighting.extra.content "" >

<!ENTITY % SVG.feSpecularLighting.element "INCLUDE" >
<![%SVG.feSpecularLighting.element;[
<!ENTITY % SVG.feSpecularLighting.content
  "(( %SVG.feDistantLight.qname; | %SVG.fePointLight.qname;
    | %SVG.feSpotLight.qname; ), ( %SVG.animate.qname; | %SVG.set.qname;
    | %SVG.animateColor.qname; %SVG.feSpecularLighting.extra.content; )*)"
>
<!ELEMENT %SVG.feSpecularLighting.qname; %SVG.feSpecularLighting.content; >
<!-- end of SVG.feSpecularLighting.element -->]]>

<!ENTITY % SVG.feSpecularLighting.attlist "INCLUDE" >
<![%SVG.feSpecularLighting.attlist;[
<!ATTLIST %SVG.feSpecularLighting.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Color.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
```

```
  lighting-color %SVGColor.datatype; #IMPLIED
  surfaceScale %Number.datatype; #IMPLIED
  specularConstant %Number.datatype; #IMPLIED
  specularExponent %Number.datatype; #IMPLIED
  kernelUnitLength %NumberOptionalNumber.datatype; #IMPLIED
>
<!-- end of SVG.feSpecularLighting.attlist -->]]>

<!-- feTile: Filter Effect Tile Element ..... -->

<!ENTITY % SVG.feTile.extra.content "" >

<!ENTITY % SVG.feTile.element "INCLUDE" >
<![%SVG.feTile.element;[
<!ENTITY % SVG.feTile.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feTile.extra.content; )*"
>
<!ELEMENT %SVG.feTile.qname; %SVG.feTile.content; >
<!-- end of SVG.feTile.element -->]]>

<!ENTITY % SVG.feTile.attlist "INCLUDE" >
<![%SVG.feTile.attlist;[
<!ATTLIST %SVG.feTile.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
>
<!-- end of SVG.feTile.attlist -->]]>

<!-- feTurbulence: Filter Effect Turbulence Element .... -->

<!ENTITY % SVG.feTurbulence.extra.content "" >

<!ENTITY % SVG.feTurbulence.element "INCLUDE" >
<![%SVG.feTurbulence.element;[
<!ENTITY % SVG.feTurbulence.content
  "( %SVG.animate.qname; | %SVG.set.qname;
    %SVG.feTurbulence.extra.content; )*"
>
<!ELEMENT %SVG.feTurbulence.qname; %SVG.feTurbulence.content; >
<!-- end of SVG.feTurbulence.element -->]]>

<!ENTITY % SVG.feTurbulence.attlist "INCLUDE" >
<![%SVG.feTurbulence.attlist;[
<!ATTLIST %SVG.feTurbulence.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitive.attrib;
```

```
  baseFrequency %NumberOptionalNumber.datatype; #IMPLIED
  numOctaves %Integer.datatype; #IMPLIED
  seed %Number.datatype; #IMPLIED
  stitchTiles ( stitch | noStitch ) 'noStitch'
  type ( fractalNoise | turbulence ) 'turbulence'
>
<!-- end of SVG.feTurbulence.attlist -->]]>

<!-- feDistantLight: Filter Effect Distant Light Element -->

<!ENTITY % SVG.feDistantLight.extra.content "" >

<!ENTITY % SVG.feDistantLight.element "INCLUDE" >
<![%SVG.feDistantLight.element;[
<!ENTITY % SVG.feDistantLight.content
  "( %SVG.animate.qname; | %SVG.set.qname;
     %SVG.feDistantLight.extra.content; )*"
>
<!ELEMENT %SVG.feDistantLight.qname; %SVG.feDistantLight.content; >
<!-- end of SVG.feDistantLight.element -->]]>

<!ENTITY % SVG.feDistantLight.attlist "INCLUDE" >
<![%SVG.feDistantLight.attlist;[
<!ATTLIST %SVG.feDistantLight.qname;
  %SVG.Core.attrib;
  azimuth %Number.datatype; #IMPLIED
  elevation %Number.datatype; #IMPLIED
>
<!-- end of SVG.feDistantLight.attlist -->]]>

<!-- fePointLight: Filter Effect Point Light Element ... -->

<!ENTITY % SVG.fePointLight.extra.content "" >

<!ENTITY % SVG.fePointLight.element "INCLUDE" >
<![%SVG.fePointLight.element;[
<!ENTITY % SVG.fePointLight.content
  "( %SVG.animate.qname; | %SVG.set.qname;
     %SVG.fePointLight.extra.content; )*"
>
<!ELEMENT %SVG.fePointLight.qname; %SVG.fePointLight.content; >
<!-- end of SVG.fePointLight.element -->]]>

<!ENTITY % SVG.fePointLight.attlist "INCLUDE" >
<![%SVG.fePointLight.attlist;[
<!ATTLIST %SVG.fePointLight.qname;
  %SVG.Core.attrib;
  x %Number.datatype; #IMPLIED
```

```
  y %Number.datatype; #IMPLIED
  z %Number.datatype; #IMPLIED
>
<!-- end of SVG.fePointLight.attlist -->]]>

<!-- feSpotLight: Filter Effect Spot Light Element ..... -->

<!ENTITY % SVG.feSpotLight.extra.content "" >

<!ENTITY % SVG.feSpotLight.element "INCLUDE" >
<![%SVG.feSpotLight.element;[
<!ENTITY % SVG.feSpotLight.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feSpotLight.extra.content; )*"
>
<!ELEMENT %SVG.feSpotLight.qname; %SVG.feSpotLight.content; >
<!-- end of SVG.feSpotLight.element -->]]>

<!ENTITY % SVG.feSpotLight.attlist "INCLUDE" >
<![%SVG.feSpotLight.attlist;[
<!ATTLIST %SVG.feSpotLight.qname;
  %SVG.Core.attrib;
  x %Number.datatype; #IMPLIED
  y %Number.datatype; #IMPLIED
  z %Number.datatype; #IMPLIED
  pointsAtX %Number.datatype; #IMPLIED
  pointsAtY %Number.datatype; #IMPLIED
  pointsAtZ %Number.datatype; #IMPLIED
  specularExponent %Number.datatype; #IMPLIED
  limitingConeAngle %Number.datatype; #IMPLIED
>
<!-- end of SVG.feSpotLight.attlist -->]]>

<!-- feFuncR: Filter Effect Function Red Element ..... -->

<!ENTITY % SVG.feFuncR.extra.content "" >

<!ENTITY % SVG.feFuncR.element "INCLUDE" >
<![%SVG.feFuncR.element;[
<!ENTITY % SVG.feFuncR.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncR.extra.content; )*"
>
<!ELEMENT %SVG.feFuncR.qname; %SVG.feFuncR.content; >
<!-- end of SVG.feFuncR.element -->]]>

<!ENTITY % SVG.feFuncR.attlist "INCLUDE" >
<![%SVG.feFuncR.attlist;[
<!ATTLIST %SVG.feFuncR.qname;
  %SVG.Core.attrib;
```

```
  type ( identity | table | discrete | linear | gamma ) #REQUIRED
  tableValues CDATA #IMPLIED
  slope %Number.datatype; #IMPLIED
  intercept %Number.datatype; #IMPLIED
  amplitude %Number.datatype; #IMPLIED
  exponent %Number.datatype; #IMPLIED
  offset %Number.datatype; #IMPLIED
>
<!-- end of SVG.feFuncR.attlist -->]]>

<!-- feFuncG: Filter Effect Function Green Element ..... -->

<!ENTITY % SVG.feFuncG.extra.content "" >

<!ENTITY % SVG.feFuncG.element "INCLUDE" >
<![%SVG.feFuncG.element;[
<!ENTITY % SVG.feFuncG.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncG.extra.content; )*"
>
<!ELEMENT %SVG.feFuncG.qname; %SVG.feFuncG.content; >
<!-- end of SVG.feFuncG.element -->]]>

<!ENTITY % SVG.feFuncG.attlist "INCLUDE" >
<![%SVG.feFuncG.attlist;[
<!ATTLIST %SVG.feFuncG.qname;
  %SVG.Core.attrib;
  type ( identity | table | discrete | linear | gamma ) #REQUIRED
  tableValues CDATA #IMPLIED
  slope %Number.datatype; #IMPLIED
  intercept %Number.datatype; #IMPLIED
  amplitude %Number.datatype; #IMPLIED
  exponent %Number.datatype; #IMPLIED
  offset %Number.datatype; #IMPLIED
>
<!-- end of SVG.feFuncG.attlist -->]]>

<!-- feFuncB: Filter Effect Function Blue Element ..... -->

<!ENTITY % SVG.feFuncB.extra.content "" >

<!ENTITY % SVG.feFuncB.element "INCLUDE" >
<![%SVG.feFuncB.element;[
<!ENTITY % SVG.feFuncB.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncB.extra.content; )*"
>
<!ELEMENT %SVG.feFuncB.qname; %SVG.feFuncB.content; >
<!-- end of SVG.feFuncB.element -->]]>
```

```
<!ENTITY % SVG.feFuncB.attlist "INCLUDE" >
<![%SVG.feFuncB.attlist;[
<!ATTLIST %SVG.feFuncB.qname;
  %SVG.Core.attrib;
  type ( identity | table | discrete | linear | gamma ) #REQUIRED
  tableValues CDATA #IMPLIED
  slope %Number.datatype; #IMPLIED
  intercept %Number.datatype; #IMPLIED
  amplitude %Number.datatype; #IMPLIED
  exponent %Number.datatype; #IMPLIED
  offset %Number.datatype; #IMPLIED
>
<!-- end of SVG.feFuncB.attlist -->]]>

<!-- feFuncA: Filter Effect Function Alpha Element ..... -->

<!ENTITY % SVG.feFuncA.extra.content "" >

<!ENTITY % SVG.feFuncA.element "INCLUDE" >
<![%SVG.feFuncA.element;[
<!ENTITY % SVG.feFuncA.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncA.extra.content; )*"
>
<!ELEMENT %SVG.feFuncA.qname; %SVG.feFuncA.content; >
<!-- end of SVG.feFuncA.element -->]]>

<!ENTITY % SVG.feFuncA.attlist "INCLUDE" >
<![%SVG.feFuncA.attlist;[
<!ATTLIST %SVG.feFuncA.qname;
  %SVG.Core.attrib;
  type ( identity | table | discrete | linear | gamma ) #REQUIRED
  tableValues CDATA #IMPLIED
  slope %Number.datatype; #IMPLIED
  intercept %Number.datatype; #IMPLIED
  amplitude %Number.datatype; #IMPLIED
  exponent %Number.datatype; #IMPLIED
  offset %Number.datatype; #IMPLIED
>
<!-- end of SVG.feFuncA.attlist -->]]>

<!-- end of svg-filter.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-filter.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
```

```
<!-- SVG 1.1 Basic Text Module ..... -->
<!-- file: svg-basic-text.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-basic-text.mod,v 1.4 2002/11/13 20:14:30 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Basic Text//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-text.mod"

   ..... -->

<!-- Basic Text

     text

   This module declares markup to provide support for text.
-->

<!-- 'font-family' property/attribute value (i.e., list of fonts) -->
<!ENTITY % FontFamilyValue.datatype "CDATA" >

<!-- 'font-size' property/attribute value -->
<!ENTITY % FontSizeValue.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.text.qname "text" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Text.class ..... -->
```

```
<!ENTITY % SVG.Text.extra.class "" >

<!ENTITY % SVG.Text.class
  "| %SVG.text.qname; %SVG.Text.extra.class;"
>

<!-- SVG.TextContent.attrib ..... -->

<!ENTITY % SVG.TextContent.extra.attrib "" >

<!ENTITY % SVG.TextContent.attrib
  "text-anchor ( start | middle | end | inherit ) #IMPLIED
   %SVG.TextContent.extra.attrib;"
>

<!-- SVG.Font.attrib ..... -->

<!ENTITY % SVG.Font.extra.attrib "" >

<!ENTITY % SVG.Font.attrib
  "font-family %FontFamilyValue.datatype; #IMPLIED
   font-size %FontSizeValue.datatype; #IMPLIED
   font-style ( normal | italic | oblique | inherit ) #IMPLIED
   font-weight ( normal | bold | bolder | lighter | 100 | 200 | 300 | 400 |
              500 | 600 | 700 | 800 | 900 | inherit ) #IMPLIED
   %SVG.Font.extra.attrib;"
>

<!-- text: Text Element ..... -->

<!ENTITY % SVG.text.extra.content "" >

<!ENTITY % SVG.text.element "INCLUDE" >
<![%SVG.text.element;[
<!ENTITY % SVG.text.content
  "( #PCDATA | %SVG.Description.class; | %SVG.Animation.class;
     %SVG.Hyperlink.class; %SVG.text.extra.content; )*"
>
<!ELEMENT %SVG.text.qname; %SVG.text.content; >
<!-- end of SVG.text.element -->]]>

<!ENTITY % SVG.text.attlist "INCLUDE" >
<![%SVG.text.attlist;[
<!ATTLIST %SVG.text.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
```

```
  %SVG.TextContent.attrib;
  %SVG.Font.attrib;
  %SVG.Paint.attrib;
  %SVG.Color.attrib;
  %SVG.Opacity.attrib;
  %SVG.Graphics.attrib;
  %SVG.Clip.attrib;
  %SVG.Mask.attrib;
  %SVG.Filter.attrib;
  %SVG.GraphicalEvents.attrib;
  %SVG.Cursor.attrib;
  %SVG.External.attrib;
  x %Coordinates.datatype; #IMPLIED
  y %Coordinates.datatype; #IMPLIED
  rotate %Numbers.datatype; #IMPLIED
  transform %TransformList.datatype; #IMPLIED
>
<!-- end of SVG.text.attlist -->]]>

<!-- end of svg-basic-text.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-basic-text.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Basic Filter Module ..... -->
<!-- file: svg-basic-filter.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-basic-filter.mod,v 1.4 2002/11/14 15:11:02 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS SVG 1.1 Basic Filter//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-filter.mod"

   ..... -->

<!-- Basic Filter

     filter, feBlend, feColorMatrix, feComponentTransfer, feComposite,
     feFlood, feGaussianBlur, feImage, feMerge, feMergeNode, feOffset,
     feTile, feFuncR, feFuncG, feFuncB, feFuncA

   This module declares markup to provide support for filter effect.
```

```
-->

<!-- 'filter' property/attribute value (e.g., 'none', <uri>) -->
<!ENTITY % FilterValue.datatype "CDATA" >

<!-- list of <number>s, but at least one and at most two -->
<!ENTITY % NumberOptionalNumber.datatype "CDATA" >

<!-- Qualified Names (Default) ..... -->

<!ENTITY % SVG.filter.qname "filter" >
<!ENTITY % SVG.feBlend.qname "feBlend" >
<!ENTITY % SVG.feColorMatrix.qname "feColorMatrix" >
<!ENTITY % SVG.feComponentTransfer.qname "feComponentTransfer" >
<!ENTITY % SVG.feComposite.qname "feComposite" >
<!ENTITY % SVG.feFlood.qname "feFlood" >
<!ENTITY % SVG.feGaussianBlur.qname "feGaussianBlur" >
<!ENTITY % SVG.feImage.qname "feImage" >
<!ENTITY % SVG.feMerge.qname "feMerge" >
<!ENTITY % SVG.feMergeNode.qname "feMergeNode" >
<!ENTITY % SVG.feOffset.qname "feOffset" >
<!ENTITY % SVG.feTile.qname "feTile" >
<!ENTITY % SVG.feFuncR.qname "feFuncR" >
<!ENTITY % SVG.feFuncG.qname "feFuncG" >
<!ENTITY % SVG.feFuncB.qname "feFuncB" >
<!ENTITY % SVG.feFuncA.qname "feFuncA" >

<!-- Attribute Collections (Default) ..... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.ColorProfile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.XLinkEmbed.attrib "" >
```

```
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Filter.class ..................................... -->

<!ENTITY % SVG.Filter.extra.class "" >

<!ENTITY % SVG.Filter.class
  "| %SVG.filter.qname; %SVG.Filter.extra.class;"
>

<!-- SVG.FilterPrimitive.class ........................... -->

<!ENTITY % SVG.FilterPrimitive.extra.class "" >

<!ENTITY % SVG.FilterPrimitive.class
  "| %SVG.feBlend.qname; | %SVG.feColorMatrix.qname;
   | %SVG.feComponentTransfer.qname; | %SVG.feComposite.qname;
   | %SVG.feFlood.qname; | %SVG.feGaussianBlur.qname; | %SVG.feImage.qname;
   | %SVG.feMerge.qname; | %SVG.feOffset.qname; | %SVG.feTile.qname;
     %SVG.FilterPrimitive.extra.class;"
>

<!-- SVG.Filter.attrib ................................... -->

<!ENTITY % SVG.Filter.extra.attrib "" >

<!ENTITY % SVG.Filter.attrib
  "filter %FilterValue.datatype; #IMPLIED
   %SVG.Filter.extra.attrib;"
>

<!-- SVG.FilterColor.attrib .............................. -->

<!ENTITY % SVG.FilterColor.extra.attrib "" >

<!ENTITY % SVG.FilterColor.attrib
  "color-interpolation-filters ( auto | sRGB | linearRGB | inherit )
                               #IMPLIED
   %SVG.FilterColor.extra.attrib;"
>

<!-- SVG.FilterPrimitive.attrib .......................... -->

<!ENTITY % SVG.FilterPrimitive.extra.attrib "" >

<!ENTITY % SVG.FilterPrimitive.attrib
  "x %Coordinate.datatype; #IMPLIED
   y %Coordinate.datatype; #IMPLIED
```

```
    width %Length.datatype; #IMPLIED
    height %Length.datatype; #IMPLIED
    result CDATA #IMPLIED
    %SVG.FilterPrimitive.extra.attrib;"
>

<!-- SVG.FilterPrimitiveWithIn.attrib ..... -->

<!ENTITY % SVG.FilterPrimitiveWithIn.extra.attrib "" >

<!ENTITY % SVG.FilterPrimitiveWithIn.attrib
  "%SVG.FilterPrimitive.attrib;
    in CDATA #IMPLIED
    %SVG.FilterPrimitiveWithIn.extra.attrib;"
>

<!-- SVG.Presentation.attrib ..... -->

<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
  "%SVG.Container.attrib;
    %SVG.Viewport.attrib;
    %SVG.Text.attrib;
    %SVG.TextContent.attrib;
    %SVG.Font.attrib;
    %SVG.Paint.attrib;
    %SVG.Color.attrib;
    %SVG.Opacity.attrib;
    %SVG.Graphics.attrib;
    %SVG.Marker.attrib;
    %SVG.ColorProfile.attrib;
    %SVG.Gradient.attrib;
    %SVG.Clip.attrib;
    %SVG.Mask.attrib;
    %SVG.Filter.attrib;
    %SVG.FilterColor.attrib;
    %SVG.Cursor.attrib;
    flood-color %SVGColor.datatype; #IMPLIED
    flood-opacity %OpacityValue.datatype; #IMPLIED
    lighting-color %SVGColor.datatype; #IMPLIED
    %SVG.Presentation.extra.attrib;"
>

<!-- filter: Filter Element ..... -->

<!ENTITY % SVG.filter.extra.content "" >
```

```
<!ENTITY % SVG.filter.element "INCLUDE" >
<![%SVG.filter.element;[
<!ENTITY % SVG.filter.content
  "(( %SVG.Description.class; )*, ( %SVG.animate.qname; | %SVG.set.qname;
     %SVG.FilterPrimitive.class; %SVG.filter.extra.content; )*)"
>
<!ELEMENT %SVG.filter.qname; %SVG.filter.content; >
<!-- end of SVG.filter.element -->]]>

<!ENTITY % SVG.filter.attlist "INCLUDE" >
<![%SVG.filter.attlist;[
<!ATTLIST %SVG.filter.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.XLink.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #IMPLIED
  height %Length.datatype; #IMPLIED
  filterRes %NumberOptionalNumber.datatype; #IMPLIED
  filterUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
  primitiveUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
>
<!-- end of SVG.filter.attlist -->]]>

<!-- feBlend: Filter Effect Blend Element ..... -->

<!ENTITY % SVG.feBlend.extra.content "" >

<!ENTITY % SVG.feBlend.element "INCLUDE" >
<![%SVG.feBlend.element;[
<!ENTITY % SVG.feBlend.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feBlend.extra.content; )*"
>
<!ELEMENT %SVG.feBlend.qname; %SVG.feBlend.content; >
<!-- end of SVG.feBlend.element -->]]>

<!ENTITY % SVG.feBlend.attlist "INCLUDE" >
<![%SVG.feBlend.attlist;[
<!ATTLIST %SVG.feBlend.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  in2 CDATA #REQUIRED
  mode ( normal | multiply | screen | darken | lighten ) 'normal'
>
```

```
<!-- end of SVG.feBlend.attlist -->]]>

<!-- feColorMatrix: Filter Effect Color Matrix Element . -->

<!ENTITY % SVG.feColorMatrix.extra.content "" >

<!ENTITY % SVG.feColorMatrix.element "INCLUDE" >
<![%SVG.feColorMatrix.element;[
<!ENTITY % SVG.feColorMatrix.content
  "( %SVG.animate.qname; | %SVG.set.qname;
    %SVG.feColorMatrix.extra.content; )*"
>
<!ELEMENT %SVG.feColorMatrix.qname; %SVG.feColorMatrix.content; >
<!-- end of SVG.feColorMatrix.element -->]]>

<!ENTITY % SVG.feColorMatrix.attlist "INCLUDE" >
<![%SVG.feColorMatrix.attlist;[
<!ATTLIST %SVG.feColorMatrix.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  type ( matrix | saturate | hueRotate | luminanceToAlpha ) 'matrix'
  values CDATA #IMPLIED
>
<!-- end of SVG.feColorMatrix.attlist -->]]>

<!-- feComponentTransfer: Filter Effect Component Transfer Element -->

<!ENTITY % SVG.feComponentTransfer.extra.content "" >

<!ENTITY % SVG.feComponentTransfer.element "INCLUDE" >
<![%SVG.feComponentTransfer.element;[
<!ENTITY % SVG.feComponentTransfer.content
  "( %SVG.feFuncR.qname;?, %SVG.feFuncG.qname;?, %SVG.feFuncB.qname;?,
    %SVG.feFuncA.qname;? %SVG.feComponentTransfer.extra.content; )"
>
<!ELEMENT %SVG.feComponentTransfer.qname; %SVG.feComponentTransfer.content; >
<!-- end of SVG.feComponentTransfer.element -->]]>

<!ENTITY % SVG.feComponentTransfer.attlist "INCLUDE" >
<![%SVG.feComponentTransfer.attlist;[
<!ATTLIST %SVG.feComponentTransfer.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
>
<!-- end of SVG.feComponentTransfer.attlist -->]]>
```

```
<!-- feComposite: Filter Effect Composite Element ..... -->

<!ENTITY % SVG.feComposite.extra.content "" >

<!ENTITY % SVG.feComposite.element "INCLUDE" >
<![%SVG.feComposite.element;[
<!ENTITY % SVG.feComposite.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feComposite.extra.content; )*"
>
<!ELEMENT %SVG.feComposite.qname; %SVG.feComposite.content; >
<!-- end of SVG.feComposite.element -->]]>

<!ENTITY % SVG.feComposite.attlist "INCLUDE" >
<![%SVG.feComposite.attlist;[
<!ATTLIST %SVG.feComposite.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  in2 CDATA #REQUIRED
  operator ( over | in | out | atop | xor | arithmetic ) 'over'
  k1 %Number.datatype; #IMPLIED
  k2 %Number.datatype; #IMPLIED
  k3 %Number.datatype; #IMPLIED
  k4 %Number.datatype; #IMPLIED
>
<!-- end of SVG.feComposite.attlist -->]]>

<!-- feFlood: Filter Effect Flood Element ..... -->

<!ENTITY % SVG.feFlood.extra.content "" >

<!ENTITY % SVG.feFlood.element "INCLUDE" >
<![%SVG.feFlood.element;[
<!ENTITY % SVG.feFlood.content
  "( %SVG.animate.qname; | %SVG.set.qname; | %SVG.animateColor.qname;
    %SVG.feFlood.extra.content; )*"
>
<!ELEMENT %SVG.feFlood.qname; %SVG.feFlood.content; >
<!-- end of SVG.feFlood.element -->]]>

<!ENTITY % SVG.feFlood.attlist "INCLUDE" >
<![%SVG.feFlood.attlist;[
<!ATTLIST %SVG.feFlood.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Color.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
```

```
  flood-color %SVGColor.datatype; #IMPLIED
  flood-opacity %OpacityValue.datatype; #IMPLIED
>
<!-- end of SVG.feFlood.attlist -->]]>

<!-- feGaussianBlur: Filter Effect Gaussian Blur Element -->

<!ENTITY % SVG.feGaussianBlur.extra.content "" >

<!ENTITY % SVG.feGaussianBlur.element "INCLUDE" >
<![%SVG.feGaussianBlur.element;[
<!ENTITY % SVG.feGaussianBlur.content
  "( %SVG.animate.qname; | %SVG.set.qname;
    %SVG.feGaussianBlur.extra.content; )*"
>
<!ELEMENT %SVG.feGaussianBlur.qname; %SVG.feGaussianBlur.content; >
<!-- end of SVG.feGaussianBlur.element -->]]>

<!ENTITY % SVG.feGaussianBlur.attlist "INCLUDE" >
<![%SVG.feGaussianBlur.attlist;[
<!ATTLIST %SVG.feGaussianBlur.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  stdDeviation %NumberOptionalNumber.datatype; #IMPLIED
>
<!-- end of SVG.feGaussianBlur.attlist -->]]>

<!-- feImage: Filter Effect Image Element ..... -->

<!ENTITY % SVG.feImage.extra.content "" >

<!ENTITY % SVG.feImage.element "INCLUDE" >
<![%SVG.feImage.element;[
<!ENTITY % SVG.feImage.content
  "( %SVG.animate.qname; | %SVG.set.qname; | %SVG.animateTransform.qname;
    %SVG.feImage.extra.content; )*"
>
<!ELEMENT %SVG.feImage.qname; %SVG.feImage.content; >
<!-- end of SVG.feImage.element -->]]>

<!ENTITY % SVG.feImage.attlist "INCLUDE" >
<![%SVG.feImage.attlist;[
<!ATTLIST %SVG.feImage.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.FilterPrimitive.attrib;
```

```
  %SVG.XLinkEmbed.attrib;
  %SVG.External.attrib;
  preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
>
<!-- end of SVG.feImage.attlist -->]]>

<!-- feMerge: Filter Effect Merge Element ..... -->

<!ENTITY % SVG.feMerge.extra.content "" >

<!ENTITY % SVG.feMerge.element "INCLUDE" >
<![%SVG.feMerge.element;[
<!ENTITY % SVG.feMerge.content
  "( %SVG.feMergeNode.qname; %SVG.feMerge.extra.content; )*"
>
<!ELEMENT %SVG.feMerge.qname; %SVG.feMerge.content; >
<!-- end of SVG.feMerge.element -->]]>

<!ENTITY % SVG.feMerge.attlist "INCLUDE" >
<![%SVG.feMerge.attlist;[
<!ATTLIST %SVG.feMerge.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitive.attrib;
>
<!-- end of SVG.feMerge.attlist -->]]>

<!-- feMergeNode: Filter Effect Merge Node Element ..... -->

<!ENTITY % SVG.feMergeNode.extra.content "" >

<!ENTITY % SVG.feMergeNode.element "INCLUDE" >
<![%SVG.feMergeNode.element;[
<!ENTITY % SVG.feMergeNode.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feMergeNode.extra.content; )*"
>
<!ELEMENT %SVG.feMergeNode.qname; %SVG.feMergeNode.content; >
<!-- end of SVG.feMergeNode.element -->]]>

<!ENTITY % SVG.feMergeNode.attlist "INCLUDE" >
<![%SVG.feMergeNode.attlist;[
<!ATTLIST %SVG.feMergeNode.qname;
  %SVG.Core.attrib;
  in CDATA #IMPLIED
>
<!-- end of SVG.feMergeNode.attlist -->]]>

<!-- feOffset: Filter Effect Offset Element ..... -->
```

```
<!ENTITY % SVG.feOffset.extra.content "" >

<!ENTITY % SVG.feOffset.element "INCLUDE" >
<![%SVG.feOffset.element;[
<!ENTITY % SVG.feOffset.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feOffset.extra.content; )*"
>
<!ELEMENT %SVG.feOffset.qname; %SVG.feOffset.content; >
<!-- end of SVG.feOffset.element -->]]>

<!ENTITY % SVG.feOffset.attlist "INCLUDE" >
<![%SVG.feOffset.attlist;[
<!ATTLIST %SVG.feOffset.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
  dx %Number.datatype; #IMPLIED
  dy %Number.datatype; #IMPLIED
>
<!-- end of SVG.feOffset.attlist -->]]>

<!-- feTile: Filter Effect Tile Element ..... -->

<!ENTITY % SVG.feTile.extra.content "" >

<!ENTITY % SVG.feTile.element "INCLUDE" >
<![%SVG.feTile.element;[
<!ENTITY % SVG.feTile.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feTile.extra.content; )*"
>
<!ELEMENT %SVG.feTile.qname; %SVG.feTile.content; >
<!-- end of SVG.feTile.element -->]]>

<!ENTITY % SVG.feTile.attlist "INCLUDE" >
<![%SVG.feTile.attlist;[
<!ATTLIST %SVG.feTile.qname;
  %SVG.Core.attrib;
  %SVG.FilterColor.attrib;
  %SVG.FilterPrimitiveWithIn.attrib;
>
<!-- end of SVG.feTile.attlist -->]]>

<!-- feFuncR: Filter Effect Function Red Element ..... -->

<!ENTITY % SVG.feFuncR.extra.content "" >

<!ENTITY % SVG.feFuncR.element "INCLUDE" >
```

```
<![%SVG.feFuncR.element;[
<!ENTITY % SVG.feFuncR.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncR.extra.content; )*"
>
<!ELEMENT %SVG.feFuncR.qname; %SVG.feFuncR.content; >
<!-- end of SVG.feFuncR.element -->]]>

<!ENTITY % SVG.feFuncR.attlist "INCLUDE" >
<![%SVG.feFuncR.attlist;[
<!ATTLIST %SVG.feFuncR.qname;
  %SVG.Core.attrib;
  type ( identity | table | discrete | linear | gamma ) #REQUIRED
  tableValues CDATA #IMPLIED
  slope %Number.datatype; #IMPLIED
  intercept %Number.datatype; #IMPLIED
  amplitude %Number.datatype; #IMPLIED
  exponent %Number.datatype; #IMPLIED
  offset %Number.datatype; #IMPLIED
>
<!-- end of SVG.feFuncR.attlist -->]]>

<!-- feFuncG: Filter Effect Function Green Element ..... -->

<!ENTITY % SVG.feFuncG.extra.content "" >

<!ENTITY % SVG.feFuncG.element "INCLUDE" >
<![%SVG.feFuncG.element;[
<!ENTITY % SVG.feFuncG.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncG.extra.content; )*"
>
<!ELEMENT %SVG.feFuncG.qname; %SVG.feFuncG.content; >
<!-- end of SVG.feFuncG.element -->]]>

<!ENTITY % SVG.feFuncG.attlist "INCLUDE" >
<![%SVG.feFuncG.attlist;[
<!ATTLIST %SVG.feFuncG.qname;
  %SVG.Core.attrib;
  type ( identity | table | discrete | linear | gamma ) #REQUIRED
  tableValues CDATA #IMPLIED
  slope %Number.datatype; #IMPLIED
  intercept %Number.datatype; #IMPLIED
  amplitude %Number.datatype; #IMPLIED
  exponent %Number.datatype; #IMPLIED
  offset %Number.datatype; #IMPLIED
>
<!-- end of SVG.feFuncG.attlist -->]]>

<!-- feFuncB: Filter Effect Function Blue Element ..... -->
```

```
<!ENTITY % SVG.feFuncB.extra.content "" >

<!ENTITY % SVG.feFuncB.element "INCLUDE" >
<![%SVG.feFuncB.element;[
<!ENTITY % SVG.feFuncB.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncB.extra.content; )*"
>
<!ELEMENT %SVG.feFuncB.qname; %SVG.feFuncB.content; >
<!-- end of SVG.feFuncB.element -->]]>

<!ENTITY % SVG.feFuncB.attlist "INCLUDE" >
<![%SVG.feFuncB.attlist;[
<!ATTLIST %SVG.feFuncB.qname;
  %SVG.Core.attrib;
  type ( identity | table | discrete | linear | gamma ) #REQUIRED
  tableValues CDATA #IMPLIED
  slope %Number.datatype; #IMPLIED
  intercept %Number.datatype; #IMPLIED
  amplitude %Number.datatype; #IMPLIED
  exponent %Number.datatype; #IMPLIED
  offset %Number.datatype; #IMPLIED
>
<!-- end of SVG.feFuncB.attlist -->]]>

<!-- feFuncA: Filter Effect Function Alpha Element ..... -->

<!ENTITY % SVG.feFuncA.extra.content "" >

<!ENTITY % SVG.feFuncA.element "INCLUDE" >
<![%SVG.feFuncA.element;[
<!ENTITY % SVG.feFuncA.content
  "( %SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncA.extra.content; )*"
>
<!ELEMENT %SVG.feFuncA.qname; %SVG.feFuncA.content; >
<!-- end of SVG.feFuncA.element -->]]>

<!ENTITY % SVG.feFuncA.attlist "INCLUDE" >
<![%SVG.feFuncA.attlist;[
<!ATTLIST %SVG.feFuncA.qname;
  %SVG.Core.attrib;
  type ( identity | table | discrete | linear | gamma ) #REQUIRED
  tableValues CDATA #IMPLIED
  slope %Number.datatype; #IMPLIED
  intercept %Number.datatype; #IMPLIED
  amplitude %Number.datatype; #IMPLIED
  exponent %Number.datatype; #IMPLIED
  offset %Number.datatype; #IMPLIED
```

>
<!-- end of SVG.feFuncA.attlist -->]]>

<!-- end of svg-basic-filter.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-basic-filter.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Structure Module ..... -->
<!-- file: xhtml-struct-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-struct-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Document Structure 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-struct-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Document Structure

      title, head, body, html

   The Structure Module defines the major structural elements and
   their attributes.

   Note that the content model of the head element type is redeclared
   when the Base Module is included in the DTD.

   The parameter entity containing the XML namespace URI value used
   for XHTML is '%XHTML.xmlns;', defined in the Qualified Names module.
-->

<!-- title: Document Title ..... -->

<!-- The title element is not considered part of the flow of text.
   It should be displayed, for example as the page header or
   window title. Exactly one title is required per document.
-->

```
<!ENTITY % title.element  "INCLUDE" >
<![%title.element;[
<!ENTITY % title.content  "( #PCDATA )" >
<!ENTITY % title.qname  "title" >
<!ELEMENT %title.qname;  %title.content; >
<!-- end of title.element -->]]>

<!ENTITY % title.attlist  "INCLUDE" >
<![%title.attlist;[
<!ATTLIST %title.qname;
    %XHTML.xmlns.attrib;
    %I18n.attrib;
>
<!-- end of title.attlist -->]]>

<!-- head: Document Head ..... -->

<!ENTITY % head.element  "INCLUDE" >
<![%head.element;[
<!ENTITY % head.content
  "( %HeadOpts.mix;, %title.qname;, %HeadOpts.mix; )"
>
<!ENTITY % head.qname  "head" >
<!ELEMENT %head.qname;  %head.content; >
<!-- end of head.element -->]]>

<!ENTITY % head.attlist  "INCLUDE" >
<![%head.attlist;[
<!-- reserved for future use with document profiles
-->
<!ENTITY % profile.attrib
   "profile      %URI.datatype;          '%XHTML.profile;'"
>

<!ATTLIST %head.qname;
    %XHTML.xmlns.attrib;
    %I18n.attrib;
    %profile.attrib;
>
<!-- end of head.attlist -->]]>

<!-- body: Document Body ..... -->

<!ENTITY % body.element  "INCLUDE" >
<![%body.element;[
<!ENTITY % body.content
   "( %Block.mix; )+"
>
```

```
<!ENTITY % body.qname  "body" >
<!ELEMENT %body.qname;  %body.content; >
<!-- end of body.element -->]]>

<!ENTITY % body.attlist  "INCLUDE" >
<![%body.attlist;[
<!ATTLIST %body.qname;
    %Common.attrib;
>
<!-- end of body.attlist -->]]>

<!-- html: XHTML Document Element ..... -->

<!ENTITY % html.element  "INCLUDE" >
<![%html.element;[
<!ENTITY % html.content  "( %head.qname;, %body.qname; )" >
<!ENTITY % html.qname  "html" >
<!ELEMENT %html.qname;  %html.content; >
<!-- end of html.element -->]]>

<!ENTITY % html.attlist  "INCLUDE" >
<![%html.attlist;[
<!-- version attribute value defined in driver
-->
<!ENTITY % XHTML.version.attrib
   "version      %FPI.datatype;          #FIXED '%XHTML.version;'"
>

<!-- see the Qualified Names module for information
   on how to extend XHTML using XML namespaces
-->
<!ATTLIST %html.qname;
    %XHTML.xmlns.attrib;
    %XHTML.version.attrib;
    %I18n.attrib;
>
<!-- end of html.attlist -->]]>

<!-- end of xhtml-struct-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-struct-1.mod
No license file was found, but licenses were detected in source scan.

dimensional graphics in XML.

   The Scalable Vector Graphics (SVG)

Copyright 2001, 2002 World Wide Web Consortium
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the SVG DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11.dtd,v 1.2 2002/04/20 18:07:43 fujisawa Exp $

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg11.dtd
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Basic Paint Attribute Module ..... -->
<!-- file: svg-basic-paint-attrib.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-basic-paint-attrib.mod,v 1.1 2002/11/13 20:14:30 fujisawa Exp $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Basic Paint Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-paint-attrib.mod"

..... -->

<!-- Basic Paint Attribute

fill, fill-rule, stroke, stroke-dasharray, stroke-dashoffset,
stroke-linecap, stroke-linejoin, stroke-miterlimit, stroke-width, color,
color-rendering

This module defines the Paint and Color attribute sets.
-->

<!-- a 'fill' or 'stroke' property/attribute value: <paint> -->
<!ENTITY % Paint.datatype "CDATA" >

```
<!-- 'stroke-dasharray' property/attribute value (e.g., 'none', list of <number>s) -->
<!ENTITY % StrokeDashArrayValue.datatype "CDATA" >

<!-- 'stroke-dashoffset' property/attribute value (e.g., 'none', <legnth>) -->
<!ENTITY % StrokeDashOffsetValue.datatype "CDATA" >

<!-- 'stroke-miterlimit' property/attribute value (e.g., <number>) -->
<!ENTITY % StrokeMiterLimitValue.datatype "CDATA" >

<!-- 'stroke-width' property/attribute value (e.g., <length>) -->
<!ENTITY % StrokeWidthValue.datatype "CDATA" >

<!ENTITY % SVG.fill.attrib
  "fill %Paint.datatype; #IMPLIED"
>

<!ENTITY % SVG.fill-rule.attrib
  "fill-rule %ClipFillRule.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke.attrib
  "stroke %Paint.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-dasharray.attrib
  "stroke-dasharray %StrokeDashArrayValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-dashoffset.attrib
  "stroke-dashoffset %StrokeDashOffsetValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-linecap.attrib
  "stroke-linecap ( butt | round | square | inherit ) #IMPLIED"
>

<!ENTITY % SVG.stroke-linejoin.attrib
  "stroke-linejoin ( miter | round | bevel | inherit ) #IMPLIED"
>

<!ENTITY % SVG.stroke-miterlimit.attrib
  "stroke-miterlimit %StrokeMiterLimitValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-width.attrib
  "stroke-width %StrokeWidthValue.datatype; #IMPLIED"
>
```

```
<!ENTITY % SVG.Paint.extra.attrib "" >

<!ENTITY % SVG.Paint.attrib
  "%SVG.fill.attrib;
   %SVG.fill-rule.attrib;
   %SVG.stroke.attrib;
   %SVG.stroke-dasharray.attrib;
   %SVG.stroke-dashoffset.attrib;
   %SVG.stroke-linecap.attrib;
   %SVG.stroke-linejoin.attrib;
   %SVG.stroke-miterlimit.attrib;
   %SVG.stroke-width.attrib;
   %SVG.Paint.extra.attrib;"
>

<!ENTITY % SVG.color.attrib
  "color %Color.datatype; #IMPLIED"
>

<!ENTITY % SVG.color-rendering.attrib
  "color-rendering ( auto | optimizeSpeed | optimizeQuality | inherit )
                #IMPLIED"
>

<!ENTITY % SVG.Color.extra.attrib "" >

<!ENTITY % SVG.Color.attrib
  "%SVG.color.attrib;
   %SVG.color-rendering.attrib;
   %SVG.Color.extra.attrib;"
>

<!-- end of svg-basic-paint-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/svg11/svg-basic-paint-attrib.mod
No license file was found, but licenses were detected in source scan.

<!--
   File isoamsr.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html
```

```
   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

   References to the VARIANT SELECTOR 1 character (&#x0FE00;)
   should match the uses listed in Unicode Technical Report 25.

   Entity names in this file are derived from files carrying the
   following notice:

   (C) International Organization for Standardization 1991
   Permission to copy in any form is granted for use with
   conforming SGML systems and applications as defined in
   ISO 8879, provided this notice is included in all copies.

-->


<!--

     Public identifier: -//W3C//ENTITIES Added Math Symbols: Relations//EN//XML
     System identifier: http://www.w3.org/2003/entities/2007/isoamsr.ent

   The public identifier should always be used verbatim.
   The system identifier may be changed to suit local requirements.

   Typical invocation:

     <!ENTITY % isoamsr PUBLIC
       "-//W3C//ENTITIES Added Math Symbols: Relations//EN//XML"
       "http://www.w3.org/2003/entities/2007/isoamsr.ent"
     >
     %isoamsr;

-->

<!ENTITY ape          "&#x0224A;" ><!--ALMOST EQUAL OR EQUAL TO -->
<!ENTITY apE          "&#x02A70;" ><!--APPROXIMATELY EQUAL OR EQUAL TO -->
<!ENTITY apid         "&#x0224B;" ><!--TRIPLE TILDE -->
<!ENTITY asymp          "&#x02248;" ><!--ALMOST EQUAL TO -->
<!ENTITY Barv         "&#x02AE7;" ><!--SHORT DOWN TACK WITH OVERBAR -->
<!ENTITY bcong          "&#x0224C;" ><!--ALL EQUAL TO -->
<!ENTITY bepsi        "&#x003F6;" ><!--GREEK REVERSED LUNATE EPSILON SYMBOL -->
<!ENTITY bowtie         "&#x022C8;" ><!--BOWTIE -->
<!ENTITY bsim         "&#x0223D;" ><!--REVERSED TILDE -->
```

```
<!ENTITY bsime          "&#x022CD;" ><!--REVERSED TILDE EQUALS -->
<!ENTITY bsolhsub       "&#x027C8;" ><!--REVERSE SOLIDUS PRECEDING SUBSET -->
<!ENTITY bump           "&#x0224E;" ><!--GEOMETRICALLY EQUIVALENT TO -->
<!ENTITY bumpe          "&#x0224F;" ><!--DIFFERENCE BETWEEN -->
<!ENTITY bumpE          "&#x02AAE;" ><!--EQUALS SIGN WITH BUMPY ABOVE -->
<!ENTITY cire          "&#x02257;" ><!--RING EQUAL TO -->
<!ENTITY Colon          "&#x02237;" ><!--PROPORTION -->
<!ENTITY colone         "&#x02254;" ><!--COLON EQUALS -->
<!ENTITY Colone         "&#x02A74;" ><!--DOUBLE COLON EQUAL -->
<!ENTITY congdot        "&#x02A6D;" ><!--CONGRUENT WITH DOT ABOVE -->
<!ENTITY csub          "&#x02ACF;" ><!--CLOSED SUBSET -->
<!ENTITY csube          "&#x02AD1;" ><!--CLOSED SUBSET OR EQUAL TO -->
<!ENTITY csup          "&#x02AD0;" ><!--CLOSED SUPERSET -->
<!ENTITY csupe          "&#x02AD2;" ><!--CLOSED SUPERSET OR EQUAL TO -->
<!ENTITY cuepr          "&#x022DE;" ><!--EQUAL TO OR PRECEDES -->
<!ENTITY cuesc          "&#x022DF;" ><!--EQUAL TO OR SUCCEEDS -->
<!ENTITY dashv          "&#x022A3;" ><!--LEFT TACK -->
<!ENTITY Dashv          "&#x02AE4;" ><!--VERTICAL BAR DOUBLE LEFT TURNSTILE -->
<!ENTITY easter         "&#x02A6E;" ><!--EQUALS WITH ASTERISK -->
<!ENTITY ecir          "&#x02256;" ><!--RING IN EQUAL TO -->
<!ENTITY ecolon         "&#x02255;" ><!--EQUALS COLON -->
<!ENTITY eDDot          "&#x02A77;" ><!--EQUALS SIGN WITH TWO DOTS ABOVE AND TWO DOTS
BELOW -->
<!ENTITY eDot           "&#x02251;" ><!--GEOMETRICALLY EQUAL TO -->
<!ENTITY efDot          "&#x02252;" ><!--APPROXIMATELY EQUAL TO OR THE IMAGE OF -->
<!ENTITY eg            "&#x02A9A;" ><!--DOUBLE-LINE EQUAL TO OR GREATER-THAN -->
<!ENTITY egs           "&#x02A96;" ><!--SLANTED EQUAL TO OR GREATER-THAN -->
<!ENTITY egsdot         "&#x02A98;" ><!--SLANTED EQUAL TO OR GREATER-THAN WITH DOT INSIDE -
->
<!ENTITY el            "&#x02A99;" ><!--DOUBLE-LINE EQUAL TO OR LESS-THAN -->
<!ENTITY els           "&#x02A95;" ><!--SLANTED EQUAL TO OR LESS-THAN -->
<!ENTITY elsdot         "&#x02A97;" ><!--SLANTED EQUAL TO OR LESS-THAN WITH DOT INSIDE -->
<!ENTITY equest         "&#x0225F;" ><!--QUESTIONED EQUAL TO -->
<!ENTITY equivDD        "&#x02A78;" ><!--EQUIVALENT WITH FOUR DOTS ABOVE -->
<!ENTITY erDot          "&#x02253;" ><!--IMAGE OF OR APPROXIMATELY EQUAL TO -->
<!ENTITY esdot          "&#x02250;" ><!--APPROACHES THE LIMIT -->
<!ENTITY esim           "&#x02242;" ><!--MINUS TILDE -->
<!ENTITY Esim           "&#x02A73;" ><!--EQUALS SIGN ABOVE TILDE OPERATOR -->
<!ENTITY fork          "&#x022D4;" ><!--PITCHFORK -->
<!ENTITY forkv          "&#x02AD9;" ><!--ELEMENT OF OPENING DOWNWARDS -->
<!ENTITY frown          "&#x02322;" ><!--FROWN -->
<!ENTITY gap           "&#x02A86;" ><!--GREATER-THAN OR APPROXIMATE -->
<!ENTITY gE            "&#x02267;" ><!--GREATER-THAN OVER EQUAL TO -->
<!ENTITY gel           "&#x022DB;" ><!--GREATER-THAN EQUAL TO OR LESS-THAN -->
<!ENTITY gEl           "&#x02A8C;" ><!--GREATER-THAN ABOVE DOUBLE-LINE EQUAL ABOVE LESS-
THAN -->
<!ENTITY ges           "&#x02A7E;" ><!--GREATER-THAN OR SLANTED EQUAL TO -->
<!ENTITY gescc          "&#x02AA9;" ><!--GREATER-THAN CLOSED BY CURVE ABOVE SLANTED
```

EQUAL -->
<!ENTITY gesdot "⪀" ><!--GREATER-THAN OR SLANTED EQUAL TO WITH DOT INSIDE -->
<!ENTITY gesdoto "⪂" ><!--GREATER-THAN OR SLANTED EQUAL TO WITH DOT ABOVE -->
<!ENTITY gesdotol "⪄" ><!--GREATER-THAN OR SLANTED EQUAL TO WITH DOT ABOVE LEFT -->
<!ENTITY gesl "⋛︀" ><!--GREATER-THAN slanted EQUAL TO OR LESS-THAN -->
<!ENTITY gesles "⪔" ><!--GREATER-THAN ABOVE SLANTED EQUAL ABOVE LESS-THAN ABOVE SLANTED EQUAL -->
<!ENTITY Gg "⋙" ><!--VERY MUCH GREATER-THAN -->
<!ENTITY gl "≷" ><!--GREATER-THAN OR LESS-THAN -->
<!ENTITY gla "⪥" ><!--GREATER-THAN BESIDE LESS-THAN -->
<!ENTITY glE "⪒" ><!--GREATER-THAN ABOVE LESS-THAN ABOVE DOUBLE-LINE EQUAL -->
<!ENTITY glj "⪤" ><!--GREATER-THAN OVERLAPPING LESS-THAN -->
<!ENTITY gsim "≳" ><!--GREATER-THAN OR EQUIVALENT TO -->
<!ENTITY gsime "⪎" ><!--GREATER-THAN ABOVE SIMILAR OR EQUAL -->
<!ENTITY gsiml "⪐" ><!--GREATER-THAN ABOVE SIMILAR ABOVE LESS-THAN -->
<!ENTITY Gt "≫" ><!--MUCH GREATER-THAN -->
<!ENTITY gtcc "⪧" ><!--GREATER-THAN CLOSED BY CURVE -->
<!ENTITY gtcir "⩺" ><!--GREATER-THAN WITH CIRCLE INSIDE -->
<!ENTITY gtdot "⋗" ><!--GREATER-THAN WITH DOT -->
<!ENTITY gtquest "⩼" ><!--GREATER-THAN WITH QUESTION MARK ABOVE -->
<!ENTITY gtrarr "⥸" ><!--GREATER-THAN ABOVE RIGHTWARDS ARROW -->
<!ENTITY homtht "∻" ><!--HOMOTHETIC -->
<!ENTITY lap "⪅" ><!--LESS-THAN OR APPROXIMATE -->
<!ENTITY lat "⪫" ><!--LARGER THAN -->
<!ENTITY late "⪭" ><!--LARGER THAN OR EQUAL TO -->
<!ENTITY lates "⪭︀" ><!--LARGER THAN OR slanted EQUAL -->
<!ENTITY lE "≦" ><!--LESS-THAN OVER EQUAL TO -->
<!ENTITY leg "⋚" ><!--LESS-THAN EQUAL TO OR GREATER-THAN -->
<!ENTITY lEg "⪋" ><!--LESS-THAN ABOVE DOUBLE-LINE EQUAL ABOVE GREATER-THAN -->
<!ENTITY les "⩽" ><!--LESS-THAN OR SLANTED EQUAL TO -->
<!ENTITY lescc "⪨" ><!--LESS-THAN CLOSED BY CURVE ABOVE SLANTED EQUAL -->
<!ENTITY lesdot "⩿" ><!--LESS-THAN OR SLANTED EQUAL TO WITH DOT INSIDE -->
<!ENTITY lesdoto "⪁" ><!--LESS-THAN OR SLANTED EQUAL TO WITH DOT ABOVE -->
<!ENTITY lesdotor "⪃" ><!--LESS-THAN OR SLANTED EQUAL TO WITH DOT ABOVE RIGHT -->
<!ENTITY lesg "⋚︀" ><!--LESS-THAN slanted EQUAL TO OR GREATER-THAN -->
<!ENTITY lesges "⪓" ><!--LESS-THAN ABOVE SLANTED EQUAL ABOVE GREATER-THAN ABOVE SLANTED EQUAL -->
<!ENTITY lg "≶" ><!--LESS-THAN OR GREATER-THAN -->
<!ENTITY lgE "⪑" ><!--LESS-THAN ABOVE GREATER-THAN ABOVE DOUBLE-LINE EQUAL -->

```
<!ENTITY Ll "&#x022D8;" ><!--VERY MUCH LESS-THAN -->
<!ENTITY lsim "&#x02272;" ><!--LESS-THAN OR EQUIVALENT TO -->
<!ENTITY lsime "&#x02A8D;" ><!--LESS-THAN ABOVE SIMILAR OR EQUAL -->
<!ENTITY lsimg "&#x02A8F;" ><!--LESS-THAN ABOVE SIMILAR ABOVE GREATER-THAN -->
<!ENTITY Lt "&#x0226A;" ><!--MUCH LESS-THAN -->
<!ENTITY ltcc "&#x02AA6;" ><!--LESS-THAN CLOSED BY CURVE -->
<!ENTITY ltcir "&#x02A79;" ><!--LESS-THAN WITH CIRCLE INSIDE -->
<!ENTITY ltdot "&#x022D6;" ><!--LESS-THAN WITH DOT -->
<!ENTITY ltlarr "&#x02976;" ><!--LESS-THAN ABOVE LEFTWARDS ARROW -->
<!ENTITY ltquest "&#x02A7B;" ><!--LESS-THAN WITH QUESTION MARK ABOVE -->
<!ENTITY ltrie "&#x022B4;" ><!--NORMAL SUBGROUP OF OR EQUAL TO -->
<!ENTITY mcomma "&#x02A29;" ><!--MINUS SIGN WITH COMMA ABOVE -->
<!ENTITY mDDot "&#x0223A;" ><!--GEOMETRIC PROPORTION -->
<!ENTITY mid "&#x02223;" ><!--DIVIDES -->
<!ENTITY mlcp "&#x02ADB;" ><!--TRANSVERSAL INTERSECTION -->
<!ENTITY models "&#x022A7;" ><!--MODELS -->
<!ENTITY mstpos "&#x0223E;" ><!--INVERTED LAZY S -->
<!ENTITY pr "&#x0227A;" ><!--PRECEDES -->
<!ENTITY Pr "&#x02ABB;" ><!--DOUBLE PRECEDES -->
<!ENTITY prap "&#x02AB7;" ><!--PRECEDES ABOVE ALMOST EQUAL TO -->
<!ENTITY prcue "&#x0227C;" ><!--PRECEDES OR EQUAL TO -->
<!ENTITY pre "&#x02AAF;" ><!--PRECEDES ABOVE SINGLE-LINE EQUALS SIGN -->
<!ENTITY prE "&#x02AB3;" ><!--PRECEDES ABOVE EQUALS SIGN -->
<!ENTITY prsim "&#x0227E;" ><!--PRECEDES OR EQUIVALENT TO -->
<!ENTITY prurel "&#x022B0;" ><!--PRECEDES UNDER RELATION -->
<!ENTITY ratio "&#x02236;" ><!--RATIO -->
<!ENTITY rtrie "&#x022B5;" ><!--CONTAINS AS NORMAL SUBGROUP OR EQUAL TO -->
<!ENTITY rtriltri "&#x029CE;" ><!--RIGHT TRIANGLE ABOVE LEFT TRIANGLE -->
<!ENTITY sc "&#x0227B;" ><!--SUCCEEDS -->
<!ENTITY Sc "&#x02ABC;" ><!--DOUBLE SUCCEEDS -->
<!ENTITY scap "&#x02AB8;" ><!--SUCCEEDS ABOVE ALMOST EQUAL TO -->
<!ENTITY sccue "&#x0227D;" ><!--SUCCEEDS OR EQUAL TO -->
<!ENTITY sce "&#x02AB0;" ><!--SUCCEEDS ABOVE SINGLE-LINE EQUALS SIGN -->
<!ENTITY scE "&#x02AB4;" ><!--SUCCEEDS ABOVE EQUALS SIGN -->
<!ENTITY scsim "&#x0227F;" ><!--SUCCEEDS OR EQUIVALENT TO -->
<!ENTITY sdote "&#x02A66;" ><!--EQUALS SIGN WITH DOT BELOW -->
<!ENTITY sfrown "&#x02322;" ><!--FROWN -->
<!ENTITY simg "&#x02A9E;" ><!--SIMILAR OR GREATER-THAN -->
<!ENTITY simgE "&#x02AA0;" ><!--SIMILAR ABOVE GREATER-THAN ABOVE EQUALS SIGN -->
<!ENTITY siml "&#x02A9D;" ><!--SIMILAR OR LESS-THAN -->
<!ENTITY simlE "&#x02A9F;" ><!--SIMILAR ABOVE LESS-THAN ABOVE EQUALS SIGN -->
<!ENTITY smid "&#x02223;" ><!--DIVIDES -->
<!ENTITY smile "&#x02323;" ><!--SMILE -->
<!ENTITY smt "&#x02AAA;" ><!--SMALLER THAN -->
<!ENTITY smte "&#x02AAC;" ><!--SMALLER THAN OR EQUAL TO -->
<!ENTITY smtes "&#x02AAC;&#x0FE00;" ><!--SMALLER THAN OR slanted EQUAL -->
<!ENTITY spar "&#x02225;" ><!--PARALLEL TO -->
<!ENTITY sqsub "&#x0228F;" ><!--SQUARE IMAGE OF -->
```

```
<!ENTITY sqsube         "&#x02291;" ><!--SQUARE IMAGE OF OR EQUAL TO -->
<!ENTITY sqsup          "&#x02290;" ><!--SQUARE ORIGINAL OF -->
<!ENTITY sqsupe         "&#x02292;" ><!--SQUARE ORIGINAL OF OR EQUAL TO -->
<!ENTITY ssmile         "&#x02323;" ><!--SMILE -->
<!ENTITY Sub           "&#x022D0;" ><!--DOUBLE SUBSET -->
<!ENTITY subE          "&#x02AC5;" ><!--SUBSET OF ABOVE EQUALS SIGN -->
<!ENTITY subedot        "&#x02AC3;" ><!--SUBSET OF OR EQUAL TO WITH DOT ABOVE -->
<!ENTITY submult        "&#x02AC1;" ><!--SUBSET WITH MULTIPLICATION SIGN BELOW -->
<!ENTITY subplus        "&#x02ABF;" ><!--SUBSET WITH PLUS SIGN BELOW -->
<!ENTITY subrarr        "&#x02979;" ><!--SUBSET ABOVE RIGHTWARDS ARROW -->
<!ENTITY subsim         "&#x02AC7;" ><!--SUBSET OF ABOVE TILDE OPERATOR -->
<!ENTITY subsub         "&#x02AD5;" ><!--SUBSET ABOVE SUBSET -->
<!ENTITY subsup         "&#x02AD3;" ><!--SUBSET ABOVE SUPERSET -->
<!ENTITY Sup           "&#x022D1;" ><!--DOUBLE SUPERSET -->
<!ENTITY supdsub        "&#x02AD8;" ><!--SUPERSET BESIDE AND JOINED BY DASH WITH SUBSET -->
<!ENTITY supE          "&#x02AC6;" ><!--SUPERSET OF ABOVE EQUALS SIGN -->
<!ENTITY supedot        "&#x02AC4;" ><!--SUPERSET OF OR EQUAL TO WITH DOT ABOVE -->
<!ENTITY suphsol        "&#x027C9;" ><!--SUPERSET PRECEDING SOLIDUS -->
<!ENTITY suphsub        "&#x02AD7;" ><!--SUPERSET BESIDE SUBSET -->
<!ENTITY suplarr        "&#x0297B;" ><!--SUPERSET ABOVE LEFTWARDS ARROW -->
<!ENTITY supmult        "&#x02AC2;" ><!--SUPERSET WITH MULTIPLICATION SIGN BELOW -->
<!ENTITY supplus        "&#x02AC0;" ><!--SUPERSET WITH PLUS SIGN BELOW -->
<!ENTITY supsim         "&#x02AC8;" ><!--SUPERSET OF ABOVE TILDE OPERATOR -->
<!ENTITY supsub         "&#x02AD4;" ><!--SUPERSET ABOVE SUBSET -->
<!ENTITY supsup         "&#x02AD6;" ><!--SUPERSET ABOVE SUPERSET -->
<!ENTITY thkap          "&#x02248;" ><!--ALMOST EQUAL TO -->
<!ENTITY thksim         "&#x0223C;" ><!--TILDE OPERATOR -->
<!ENTITY topfork        "&#x02ADA;" ><!--PITCHFORK WITH TEE TOP -->
<!ENTITY trie          "&#x0225C;" ><!--DELTA EQUAL TO -->
<!ENTITY twixt          "&#x0226C;" ><!--BETWEEN -->
<!ENTITY vBar           "&#x02AE8;" ><!--SHORT UP TACK WITH UNDERBAR -->
<!ENTITY Vbar           "&#x02AEB;" ><!--DOUBLE UP TACK -->
<!ENTITY vBarv          "&#x02AE9;" ><!--SHORT UP TACK ABOVE SHORT DOWN TACK -->
<!ENTITY vdash          "&#x022A2;" ><!--RIGHT TACK -->
<!ENTITY vDash          "&#x022A8;" ><!--TRUE -->
<!ENTITY Vdash          "&#x022A9;" ><!--FORCES -->
<!ENTITY VDash          "&#x022AB;" ><!--DOUBLE VERTICAL BAR DOUBLE RIGHT TURNSTILE -->
<!ENTITY Vdashl         "&#x02AE6;" ><!--LONG DASH FROM LEFT MEMBER OF DOUBLE VERTICAL --
>
<!ENTITY vltri         "&#x022B2;" ><!--NORMAL SUBGROUP OF -->
<!ENTITY vprop          "&#x0221D;" ><!--PROPORTIONAL TO -->
<!ENTITY vrtri         "&#x022B3;" ><!--CONTAINS AS NORMAL SUBGROUP -->
<!ENTITY Vvdash         "&#x022AA;" ><!--TRIPLE VERTICAL BAR RIGHT TURNSTILE -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isoamsr.ent

No license file was found, but licenses were detected in source scan.

```
<!--
   File isomopf.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

   Entity names in this file are derived from files carrying the
   following notice:

   (C) International Organization for Standardization 1991
   Permission to copy in any form is granted for use with
   conforming SGML systems and applications as defined in
   ISO 8879, provided this notice is included in all copies.

-->


<!--

     Public identifier: -//W3C//ENTITIES Math Alphabets: Open Face//EN//XML
     System identifier: http://www.w3.org/2003/entities/2007/isomopf.ent

   The public identifier should always be used verbatim.
   The system identifier may be changed to suit local requirements.

   Typical invocation:

     <!ENTITY % isomopf PUBLIC
       "-//W3C//ENTITIES Math Alphabets: Open Face//EN//XML"
       "http://www.w3.org/2003/entities/2007/isomopf.ent"
     >
     %isomopf;

-->

<!ENTITY Aopf           "&#x1D538;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL A -->
```

```
<!ENTITY Bopf          "&#x1D539;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL B -->
<!ENTITY Copf          "&#x02102;" ><!--DOUBLE-STRUCK CAPITAL C -->
<!ENTITY Dopf          "&#x1D53B;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL D -->
<!ENTITY Eopf          "&#x1D53C;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL E -->
<!ENTITY Fopf          "&#x1D53D;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL F -->
<!ENTITY Gopf          "&#x1D53E;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL G -->
<!ENTITY Hopf          "&#x0210D;" ><!--DOUBLE-STRUCK CAPITAL H -->
<!ENTITY Iopf          "&#x1D540;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL I -->
<!ENTITY Jopf          "&#x1D541;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL J -->
<!ENTITY Kopf          "&#x1D542;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL K -->
<!ENTITY Lopf          "&#x1D543;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL L -->
<!ENTITY Mopf          "&#x1D544;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL M -->
<!ENTITY Nopf          "&#x02115;" ><!--DOUBLE-STRUCK CAPITAL N -->
<!ENTITY Oopf          "&#x1D546;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL O -->
<!ENTITY Popf          "&#x02119;" ><!--DOUBLE-STRUCK CAPITAL P -->
<!ENTITY Qopf          "&#x0211A;" ><!--DOUBLE-STRUCK CAPITAL Q -->
<!ENTITY Ropf          "&#x0211D;" ><!--DOUBLE-STRUCK CAPITAL R -->
<!ENTITY Sopf          "&#x1D54A;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL S -->
<!ENTITY Topf          "&#x1D54B;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL T -->
<!ENTITY Uopf          "&#x1D54C;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL U -->
<!ENTITY Vopf          "&#x1D54D;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL V -->
<!ENTITY Wopf          "&#x1D54E;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL W -->
<!ENTITY Xopf          "&#x1D54F;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL X -->
<!ENTITY Yopf          "&#x1D550;" ><!--MATHEMATICAL DOUBLE-STRUCK CAPITAL Y -->
<!ENTITY Zopf          "&#x02124;" ><!--DOUBLE-STRUCK CAPITAL Z -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isomopf.ent
No license file was found, but licenses were detected in source scan.

```
<!--
   File isonum.ent produced by the XSL script entities.xsl
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

   Entity names in this file are derived from files carrying the
```

following notice:

(C) International Organization for Standardization 1986
Permission to copy in any form is granted for use with
conforming SGML systems and applications as defined in
ISO 8879, provided this notice is included in all copies.

-->

<!--

Public identifier: -//W3C//ENTITIES Numeric and Special Graphic//EN//XML
System identifier: http://www.w3.org/2003/entities/2007/isonum.ent

The public identifier should always be used verbatim.
The system identifier may be changed to suit local requirements.

Typical invocation:

<!ENTITY % isonum PUBLIC
"-//W3C//ENTITIES Numeric and Special Graphic//EN//XML"
"http://www.w3.org/2003/entities/2007/isonum.ent"
>
%isonum;

-->

<!ENTITY amp "&" ><!--AMPERSAND -->
<!ENTITY apos "'" ><!--APOSTROPHE -->
<!ENTITY ast "*" ><!--ASTERISK -->
<!ENTITY brvbar "¦" ><!--BROKEN BAR -->
<!ENTITY bsol "\" ><!--REVERSE SOLIDUS -->
<!ENTITY cent "¢" ><!--CENT SIGN -->
<!ENTITY colon ":" ><!--COLON -->
<!ENTITY comma "," ><!--COMMA -->
<!ENTITY commat "@" ><!--COMMERCIAL AT -->
<!ENTITY copy "©" ><!--COPYRIGHT SIGN -->
<!ENTITY curren "¤" ><!--CURRENCY SIGN -->
<!ENTITY darr "↓" ><!--DOWNWARDS ARROW -->
<!ENTITY deg "°" ><!--DEGREE SIGN -->
<!ENTITY divide "÷" ><!--DIVISION SIGN -->
<!ENTITY dollar "$" ><!--DOLLAR SIGN -->
<!ENTITY equals "=" ><!--EQUALS SIGN -->
<!ENTITY excl "!" ><!--EXCLAMATION MARK -->
<!ENTITY frac12 "½" ><!--VULGAR FRACTION ONE HALF -->
<!ENTITY frac14 "¼" ><!--VULGAR FRACTION ONE QUARTER -->
<!ENTITY frac18 "⅛" ><!--VULGAR FRACTION ONE EIGHTH -->

```
<!ENTITY frac34         "&#x000BE;" ><!--VULGAR FRACTION THREE QUARTERS -->
<!ENTITY frac38         "&#x0215C;" ><!--VULGAR FRACTION THREE EIGHTHS -->
<!ENTITY frac58         "&#x0215D;" ><!--VULGAR FRACTION FIVE EIGHTHS -->
<!ENTITY frac78         "&#x0215E;" ><!--VULGAR FRACTION SEVEN EIGHTHS -->
<!ENTITY gt            "&#x0003E;" ><!--GREATER-THAN SIGN -->
<!ENTITY half          "&#x000BD;" ><!--VULGAR FRACTION ONE HALF -->
<!ENTITY horbar         "&#x02015;" ><!--HORIZONTAL BAR -->
<!ENTITY hyphen         "&#x02010;" ><!--HYPHEN -->
<!ENTITY iexcl         "&#x000A1;" ><!--INVERTED EXCLAMATION MARK -->
<!ENTITY iquest         "&#x000BF;" ><!--INVERTED QUESTION MARK -->
<!ENTITY laquo         "&#x000AB;" ><!--LEFT-POINTING DOUBLE ANGLE QUOTATION MARK -->
<!ENTITY larr          "&#x02190;" ><!--LEFTWARDS ARROW -->
<!ENTITY lcub          "&#x0007B;" ><!--LEFT CURLY BRACKET -->
<!ENTITY ldquo         "&#x0201C;" ><!--LEFT DOUBLE QUOTATION MARK -->
<!ENTITY lowbar         "&#x0005F;" ><!--LOW LINE -->
<!ENTITY lpar          "&#x00028;" ><!--LEFT PARENTHESIS -->
<!ENTITY lsqb          "&#x0005B;" ><!--LEFT SQUARE BRACKET -->
<!ENTITY lsquo         "&#x02018;" ><!--LEFT SINGLE QUOTATION MARK -->
<!ENTITY lt            "&#38;#60;" ><!--LESS-THAN SIGN -->
<!ENTITY micro          "&#x000B5;" ><!--MICRO SIGN -->
<!ENTITY middot         "&#x000B7;" ><!--MIDDLE DOT -->
<!ENTITY nbsp          " " ><!--NO-BREAK SPACE -->
<!ENTITY not           "&#x000AC;" ><!--NOT SIGN -->
<!ENTITY num           "&#x00023;" ><!--NUMBER SIGN -->
<!ENTITY ohm           "&#x003A9;" ><!--GREEK CAPITAL LETTER OMEGA -->
<!ENTITY ordf          "&#x000AA;" ><!--FEMININE ORDINAL INDICATOR -->
<!ENTITY ordm          "&#x000BA;" ><!--MASCULINE ORDINAL INDICATOR -->
<!ENTITY para          "&#x000B6;" ><!--PILCROW SIGN -->
<!ENTITY percnt         "&#x00025;" ><!--PERCENT SIGN -->
<!ENTITY period         "&#x0002E;" ><!--FULL STOP -->
<!ENTITY plus          "&#x0002B;" ><!--PLUS SIGN -->
<!ENTITY plusmn         "&#x000B1;" ><!--PLUS-MINUS SIGN -->
<!ENTITY pound          "&#x000A3;" ><!--POUND SIGN -->
<!ENTITY quest         "&#x0003F;" ><!--QUESTION MARK -->
<!ENTITY quot          "&#x00022;" ><!--QUOTATION MARK -->
<!ENTITY raquo         "&#x000BB;" ><!--RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK -->
<!ENTITY rarr          "&#x02192;" ><!--RIGHTWARDS ARROW -->
<!ENTITY rcub          "&#x0007D;" ><!--RIGHT CURLY BRACKET -->
<!ENTITY rdquo         "&#x0201D;" ><!--RIGHT DOUBLE QUOTATION MARK -->
<!ENTITY reg           "&#x000AE;" ><!--REGISTERED SIGN -->
<!ENTITY rpar          "&#x00029;" ><!--RIGHT PARENTHESIS -->
<!ENTITY rsqb          "&#x0005D;" ><!--RIGHT SQUARE BRACKET -->
<!ENTITY rsquo         "&#x02019;" ><!--RIGHT SINGLE QUOTATION MARK -->
<!ENTITY sect          "&#x000A7;" ><!--SECTION SIGN -->
<!ENTITY semi          "&#x0003B;" ><!--SEMICOLON -->
<!ENTITY shy           "&#x000AD;" ><!--SOFT HYPHEN -->
<!ENTITY sol           "&#x0002F;" ><!--SOLIDUS -->
<!ENTITY sung          "&#x0266A;" ><!--EIGHTH NOTE -->
```

```
<!ENTITY sup1           "&#x000B9;" ><!--SUPERSCRIPT ONE -->
<!ENTITY sup2           "&#x000B2;" ><!--SUPERSCRIPT TWO -->
<!ENTITY sup3           "&#x000B3;" ><!--SUPERSCRIPT THREE -->
<!ENTITY times          "&#x000D7;" ><!--MULTIPLICATION SIGN -->
<!ENTITY trade          "&#x02122;" ><!--TRADE MARK SIGN -->
<!ENTITY uarr           "&#x02191;" ><!--UPWARDS ARROW -->
<!ENTITY verbar          "&#x0007C;" ><!--VERTICAL LINE -->
<!ENTITY yen            "&#x000A5;" ><!--YEN SIGN -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isonum.ent
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Hypertext Module  ..................................... -->
<!-- file: xhtml-hypertext-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-hypertext-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Hypertext 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-hypertext-1.mod"

   Revisions:
   (none)
   ..................................................................... -->

<!-- Hypertext

      a

   This module declares the anchor ('a') element type, which
   defines the source of a hypertext link. The destination
   (or link 'target') is identified via its 'id' attribute
   rather than the 'name' attribute as was used in HTML.
-->

<!-- .............  Anchor Element  ............. -->

<!ENTITY % a.element  "INCLUDE" >
<![%a.element;[
<!ENTITY % a.content
   "( #PCDATA | %InlNoAnchor.mix; )*"
>
```

```
<!ENTITY % a.qname  "a" >
<!ELEMENT %a.qname;  %a.content; >
<!-- end of a.element -->]]>

<!ENTITY % a.attlist  "INCLUDE" >
<![%a.attlist;[
<!ATTLIST %a.qname;
    %Common.attrib;
    href        %URI.datatype;          #IMPLIED
    charset     %Charset.datatype;      #IMPLIED
    type        %ContentType.datatype;  #IMPLIED
    hreflang    %LanguageCode.datatype; #IMPLIED
    rel         %LinkTypes.datatype;    #IMPLIED
    rev         %LinkTypes.datatype;    #IMPLIED
    accesskey   %Character.datatype;    #IMPLIED
    tabindex    %Number.datatype;       #IMPLIED
>
<!-- end of a.attlist -->]]>

<!-- end of xhtml-hypertext-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-hypertext-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Inline Structural Module  ....................................... -->
<!-- file: xhtml-inlstruct-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-inlstruct-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Inline Structural 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-inlstruct-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Inline Structural

      br, span

   This module declares the elements and their attributes
```

```
   used to support inline-level structural markup.
-->

<!-- br: forced line break ..... -->

<!ENTITY % br.element  "INCLUDE" >
<![%br.element;[

<!ENTITY % br.content  "EMPTY" >
<!ENTITY % br.qname  "br" >
<!ELEMENT %br.qname;  %br.content; >

<!-- end of br.element -->]]>

<!ENTITY % br.attlist  "INCLUDE" >
<![%br.attlist;[
<!ATTLIST %br.qname;
    %Core.attrib;
>
<!-- end of br.attlist -->]]>

<!-- span: generic inline container ..... -->

<!ENTITY % span.element  "INCLUDE" >
<![%span.element;[
<!ENTITY % span.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % span.qname  "span" >
<!ELEMENT %span.qname;  %span.content; >
<!-- end of span.element -->]]>

<!ENTITY % span.attlist  "INCLUDE" >
<![%span.attlist;[
<!ATTLIST %span.qname;
    %Common.attrib;
>
<!-- end of span.attlist -->]]>

<!-- end of xhtml-inlstruct-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-inlstruct-1.mod
No license file was found, but licenses were detected in source scan.

<!--
   File isomscr.ent produced by the XSL script entities.xsl
```

```
   from input data in unicode.xml.

   Copyright 1998 - 2009 W3C.

   Use and distribution of this code are permitted under the terms of the
   W3C Software Notice and License.
   http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231.html

   Please report any errors to David Carlisle
   via the public W3C list www-math@w3.org.

   The numeric character values assigned to each entity
   (should) match the Unicode assignments in Unicode 5.x.

   Entity names in this file are derived from files carrying the
   following notice:

   (C) International Organization for Standardization 1991
   Permission to copy in any form is granted for use with
   conforming SGML systems and applications as defined in
   ISO 8879, provided this notice is included in all copies.

-->


<!--

     Public identifier: -//W3C//ENTITIES Math Alphabets: Script//EN//XML
     System identifier: http://www.w3.org/2003/entities/2007/isomscr.ent

   The public identifier should always be used verbatim.
   The system identifier may be changed to suit local requirements.

   Typical invocation:

     <!ENTITY % isomscr PUBLIC
       "-//W3C//ENTITIES Math Alphabets: Script//EN//XML"
       "http://www.w3.org/2003/entities/2007/isomscr.ent"
     >
     %isomscr;

-->

<!ENTITY ascr          "&#x1D4B6;" ><!--MATHEMATICAL SCRIPT SMALL A -->
<!ENTITY Ascr          "&#x1D49C;" ><!--MATHEMATICAL SCRIPT CAPITAL A -->
<!ENTITY bscr          "&#x1D4B7;" ><!--MATHEMATICAL SCRIPT SMALL B -->
<!ENTITY Bscr          "&#x0212C;" ><!--SCRIPT CAPITAL B -->
<!ENTITY cscr          "&#x1D4B8;" ><!--MATHEMATICAL SCRIPT SMALL C -->
```

```
<!ENTITY Cscr          "&#x1D49E;" ><!--MATHEMATICAL SCRIPT CAPITAL C -->
<!ENTITY dscr          "&#x1D4B9;" ><!--MATHEMATICAL SCRIPT SMALL D -->
<!ENTITY Dscr          "&#x1D49F;" ><!--MATHEMATICAL SCRIPT CAPITAL D -->
<!ENTITY escr          "&#x0212F;" ><!--SCRIPT SMALL E -->
<!ENTITY Escr          "&#x02130;" ><!--SCRIPT CAPITAL E -->
<!ENTITY fscr          "&#x1D4BB;" ><!--MATHEMATICAL SCRIPT SMALL F -->
<!ENTITY Fscr          "&#x02131;" ><!--SCRIPT CAPITAL F -->
<!ENTITY gscr          "&#x0210A;" ><!--SCRIPT SMALL G -->
<!ENTITY Gscr          "&#x1D4A2;" ><!--MATHEMATICAL SCRIPT CAPITAL G -->
<!ENTITY hscr          "&#x1D4BD;" ><!--MATHEMATICAL SCRIPT SMALL H -->
<!ENTITY Hscr          "&#x0210B;" ><!--SCRIPT CAPITAL H -->
<!ENTITY iscr          "&#x1D4BE;" ><!--MATHEMATICAL SCRIPT SMALL I -->
<!ENTITY Iscr          "&#x02110;" ><!--SCRIPT CAPITAL I -->
<!ENTITY jscr          "&#x1D4BF;" ><!--MATHEMATICAL SCRIPT SMALL J -->
<!ENTITY Jscr          "&#x1D4A5;" ><!--MATHEMATICAL SCRIPT CAPITAL J -->
<!ENTITY kscr          "&#x1D4C0;" ><!--MATHEMATICAL SCRIPT SMALL K -->
<!ENTITY Kscr          "&#x1D4A6;" ><!--MATHEMATICAL SCRIPT CAPITAL K -->
<!ENTITY lscr          "&#x1D4C1;" ><!--MATHEMATICAL SCRIPT SMALL L -->
<!ENTITY Lscr          "&#x02112;" ><!--SCRIPT CAPITAL L -->
<!ENTITY mscr          "&#x1D4C2;" ><!--MATHEMATICAL SCRIPT SMALL M -->
<!ENTITY Mscr          "&#x02133;" ><!--SCRIPT CAPITAL M -->
<!ENTITY nscr          "&#x1D4C3;" ><!--MATHEMATICAL SCRIPT SMALL N -->
<!ENTITY Nscr          "&#x1D4A9;" ><!--MATHEMATICAL SCRIPT CAPITAL N -->
<!ENTITY oscr          "&#x02134;" ><!--SCRIPT SMALL O -->
<!ENTITY Oscr          "&#x1D4AA;" ><!--MATHEMATICAL SCRIPT CAPITAL O -->
<!ENTITY pscr          "&#x1D4C5;" ><!--MATHEMATICAL SCRIPT SMALL P -->
<!ENTITY Pscr          "&#x1D4AB;" ><!--MATHEMATICAL SCRIPT CAPITAL P -->
<!ENTITY qscr          "&#x1D4C6;" ><!--MATHEMATICAL SCRIPT SMALL Q -->
<!ENTITY Qscr          "&#x1D4AC;" ><!--MATHEMATICAL SCRIPT CAPITAL Q -->
<!ENTITY rscr          "&#x1D4C7;" ><!--MATHEMATICAL SCRIPT SMALL R -->
<!ENTITY Rscr          "&#x0211B;" ><!--SCRIPT CAPITAL R -->
<!ENTITY sscr          "&#x1D4C8;" ><!--MATHEMATICAL SCRIPT SMALL S -->
<!ENTITY Sscr          "&#x1D4AE;" ><!--MATHEMATICAL SCRIPT CAPITAL S -->
<!ENTITY tscr          "&#x1D4C9;" ><!--MATHEMATICAL SCRIPT SMALL T -->
<!ENTITY Tscr          "&#x1D4AF;" ><!--MATHEMATICAL SCRIPT CAPITAL T -->
<!ENTITY uscr          "&#x1D4CA;" ><!--MATHEMATICAL SCRIPT SMALL U -->
<!ENTITY Uscr          "&#x1D4B0;" ><!--MATHEMATICAL SCRIPT CAPITAL U -->
<!ENTITY vscr          "&#x1D4CB;" ><!--MATHEMATICAL SCRIPT SMALL V -->
<!ENTITY Vscr          "&#x1D4B1;" ><!--MATHEMATICAL SCRIPT CAPITAL V -->
<!ENTITY wscr          "&#x1D4CC;" ><!--MATHEMATICAL SCRIPT SMALL W -->
<!ENTITY Wscr          "&#x1D4B2;" ><!--MATHEMATICAL SCRIPT CAPITAL W -->
<!ENTITY xscr          "&#x1D4CD;" ><!--MATHEMATICAL SCRIPT SMALL X -->
<!ENTITY Xscr          "&#x1D4B3;" ><!--MATHEMATICAL SCRIPT CAPITAL X -->
<!ENTITY yscr          "&#x1D4CE;" ><!--MATHEMATICAL SCRIPT SMALL Y -->
<!ENTITY Yscr          "&#x1D4B4;" ><!--MATHEMATICAL SCRIPT CAPITAL Y -->
<!ENTITY zscr          "&#x1D4CF;" ><!--MATHEMATICAL SCRIPT SMALL Z -->
<!ENTITY Zscr          "&#x1D4B5;" ><!--MATHEMATICAL SCRIPT CAPITAL Z -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/mathml/mathml3/isomscr.ent
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Forms Module  ..................................................... -->
<!-- file: xhtml-form-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-form-1.mod,v 4.1 2001/04/10 09:42:30 altheim Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Forms 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-form-1.mod"

   Revisions:
   (none)
   ....................................................................... -->

<!-- Forms

      form, label, input, select, optgroup, option,
      textarea, fieldset, legend, button

   This module declares markup to provide support for online
   forms, based on the features found in HTML 4 forms.
-->

<!-- declare qualified element type names:
-->
<!ENTITY % form.qname  "form" >
<!ENTITY % label.qname  "label" >
<!ENTITY % input.qname  "input" >
<!ENTITY % select.qname  "select" >
<!ENTITY % optgroup.qname  "optgroup" >
<!ENTITY % option.qname  "option" >
<!ENTITY % textarea.qname  "textarea" >
<!ENTITY % fieldset.qname  "fieldset" >
<!ENTITY % legend.qname  "legend" >
<!ENTITY % button.qname  "button" >

<!-- %BlkNoForm.mix; includes all non-form block elements,
   plus %Misc.class;
-->
<!ENTITY % BlkNoForm.mix
```

```
   "%Heading.class;
    | %List.class;
    | %BlkStruct.class;
    %BlkPhras.class;
    %BlkPres.class;
    %Table.class;
    %Block.extra;
    %Misc.class;"
>

<!-- form: Form Element ..... -->

<!ENTITY % form.element  "INCLUDE" >
<![%form.element;[
<!ENTITY % form.content
   "( %BlkNoForm.mix;
    | %fieldset.qname; )+"
>
<!ELEMENT %form.qname;  %form.content; >
<!-- end of form.element -->]]>

<!ENTITY % form.attlist  "INCLUDE" >
<![%form.attlist;[
<!ATTLIST %form.qname;
    %Common.attrib;
    action       %URI.datatype;          #REQUIRED
    method       ( get | post )          'get'
    enctype      %ContentType.datatype;   'application/x-www-form-urlencoded'
    accept-charset %Charsets.datatype;    #IMPLIED
    accept       %ContentTypes.datatype;  #IMPLIED
>
<!-- end of form.attlist -->]]>

<!-- label: Form Field Label Text ..... -->

<!-- Each label must not contain more than ONE field
-->

<!ENTITY % label.element  "INCLUDE" >
<![%label.element;[
<!ENTITY % label.content
   "( #PCDATA
    | %input.qname; | %select.qname; | %textarea.qname; | %button.qname;
    | %InlStruct.class;
    %InlPhras.class;
    %I18n.class;
    %InlPres.class;
    %Anchor.class;
```

```
    %InlSpecial.class;
    %Inline.extra;
    %Misc.class; )*"
>
<!ELEMENT %label.qname;  %label.content; >
<!-- end of label.element -->]]>

<!ENTITY % label.attlist  "INCLUDE" >
<![%label.attlist;[
<!ATTLIST %label.qname;
    %Common.attrib;
    for         IDREF                #IMPLIED
    accesskey   %Character.datatype;    #IMPLIED
>
<!-- end of label.attlist -->]]>

<!-- input: Form Control ..... -->

<!ENTITY % input.element  "INCLUDE" >
<![%input.element;[
<!ENTITY % input.content  "EMPTY" >
<!ELEMENT %input.qname;  %input.content; >
<!-- end of input.element -->]]>

<!ENTITY % input.attlist  "INCLUDE" >
<![%input.attlist;[
<!ENTITY % InputType.class
   "( text | password | checkbox | radio | submit
    | reset | file | hidden | image | button )"
>
<!-- attribute 'name' required for all but submit & reset
-->
<!ATTLIST %input.qname;
    %Common.attrib;
    type        %InputType.class;       'text'
    name        CDATA                   #IMPLIED
    value       CDATA                   #IMPLIED
    checked     ( checked )             #IMPLIED
    disabled    ( disabled )            #IMPLIED
    readonly    ( readonly )            #IMPLIED
    size        %Number.datatype;       #IMPLIED
    maxlength   %Number.datatype;       #IMPLIED
    src         %URI.datatype;          #IMPLIED
    alt         %Text.datatype;         #IMPLIED
    tabindex    %Number.datatype;       #IMPLIED
    accesskey   %Character.datatype;    #IMPLIED
    accept      %ContentTypes.datatype;  #IMPLIED
>
```

```
<!-- end of input.attlist -->]]>

<!-- select: Option Selector ..... -->

<!ENTITY % select.element  "INCLUDE" >
<![%select.element;[
<!ENTITY % select.content
   "( %optgroup.qname; | %option.qname; )+"
>
<!ELEMENT %select.qname;  %select.content; >
<!-- end of select.element -->]]>

<!ENTITY % select.attlist  "INCLUDE" >
<![%select.attlist;[
<!ATTLIST %select.qname;
    %Common.attrib;
    name        CDATA                #IMPLIED
    size        %Number.datatype;      #IMPLIED
    multiple    ( multiple )         #IMPLIED
    disabled    ( disabled )         #IMPLIED
    tabindex    %Number.datatype;      #IMPLIED
>
<!-- end of select.attlist -->]]>

<!-- optgroup: Option Group ..... -->

<!ENTITY % optgroup.element  "INCLUDE" >
<![%optgroup.element;[
<!ENTITY % optgroup.content  "( %option.qname; )+" >
<!ELEMENT %optgroup.qname;  %optgroup.content; >
<!-- end of optgroup.element -->]]>

<!ENTITY % optgroup.attlist  "INCLUDE" >
<![%optgroup.attlist;[
<!ATTLIST %optgroup.qname;
    %Common.attrib;
    disabled    ( disabled )         #IMPLIED
    label       %Text.datatype;        #REQUIRED
>
<!-- end of optgroup.attlist -->]]>

<!-- option: Selectable Choice ..... -->

<!ENTITY % option.element  "INCLUDE" >
<![%option.element;[
<!ENTITY % option.content  "( #PCDATA )" >
<!ELEMENT %option.qname;  %option.content; >
<!-- end of option.element -->]]>
```

```
<!ENTITY % option.attlist  "INCLUDE" >
<![%option.attlist;[
<!ATTLIST %option.qname;
    %Common.attrib;
    selected     ( selected )            #IMPLIED
    disabled     ( disabled )            #IMPLIED
    label        %Text.datatype;         #IMPLIED
    value        CDATA                   #IMPLIED
>
<!-- end of option.attlist -->]]>

<!-- textarea: Multi-Line Text Field ..... -->

<!ENTITY % textarea.element  "INCLUDE" >
<![%textarea.element;[
<!ENTITY % textarea.content  "( #PCDATA )" >
<!ELEMENT %textarea.qname;  %textarea.content; >
<!-- end of textarea.element -->]]>

<!ENTITY % textarea.attlist  "INCLUDE" >
<![%textarea.attlist;[
<!ATTLIST %textarea.qname;
    %Common.attrib;
    name         CDATA                   #IMPLIED
    rows         %Number.datatype;       #REQUIRED
    cols         %Number.datatype;       #REQUIRED
    disabled     ( disabled )            #IMPLIED
    readonly     ( readonly )            #IMPLIED
    tabindex     %Number.datatype;       #IMPLIED
    accesskey    %Character.datatype;    #IMPLIED
>
<!-- end of textarea.attlist -->]]>

<!-- fieldset: Form Control Group ..... -->

<!-- #PCDATA is to solve the mixed content problem,
   per specification only whitespace is allowed
-->

<!ENTITY % fieldset.element  "INCLUDE" >
<![%fieldset.element;[
<!ENTITY % fieldset.content
   "( #PCDATA | %legend.qname; | %Flow.mix; )*"
>
<!ELEMENT %fieldset.qname;  %fieldset.content; >
<!-- end of fieldset.element -->]]>
```

```
<!ENTITY % fieldset.attlist  "INCLUDE" >
<![%fieldset.attlist;[
<!ATTLIST %fieldset.qname;
    %Common.attrib;
>
<!-- end of fieldset.attlist -->]]>

<!-- legend: Fieldset Legend ..... -->

<!ENTITY % legend.element  "INCLUDE" >
<![%legend.element;[
<!ENTITY % legend.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ELEMENT %legend.qname;  %legend.content; >
<!-- end of legend.element -->]]>

<!ENTITY % legend.attlist  "INCLUDE" >
<![%legend.attlist;[
<!ATTLIST %legend.qname;
    %Common.attrib;
    accesskey    %Character.datatype;     #IMPLIED
>
<!-- end of legend.attlist -->]]>

<!-- button: Push Button ..... -->

<!ENTITY % button.element  "INCLUDE" >
<![%button.element;[
<!ENTITY % button.content
   "( #PCDATA
    | %BlkNoForm.mix;
    | %InlStruct.class;
    %InlPhras.class;
    %InlPres.class;
    %I18n.class;
    %InlSpecial.class;
    %Inline.extra; )*"
>
<!ELEMENT %button.qname;  %button.content; >
<!-- end of button.element -->]]>

<!ENTITY % button.attlist  "INCLUDE" >
<![%button.attlist;[
<!ATTLIST %button.qname;
    %Common.attrib;
    name        CDATA                #IMPLIED
    value       CDATA                #IMPLIED
```

```
    type        ( button | submit | reset ) 'submit'
    disabled    ( disabled )            #IMPLIED
    tabindex    %Number.datatype;       #IMPLIED
    accesskey   %Character.datatype;    #IMPLIED
>
<!-- end of button.attlist -->]]>

<!-- end of xhtml-form-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/xhtml11/xhtml-form-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..................................................................... -->
<!-- XHTML Block Phrasal Module  ..... -->
<!-- file: xhtml-blkphras-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-blkphras-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Block Phrasal 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-blkphras-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Block Phrasal

      address, blockquote, pre, h1, h2, h3, h4, h5, h6

   This module declares the elements and their attributes used to
   support block-level phrasal markup.
-->

<!ENTITY % address.element  "INCLUDE" >
<![%address.element;[
<!ENTITY % address.content
   "( #PCDATA | %Inline.mix; )*" >
<!ENTITY % address.qname  "address" >
<!ELEMENT %address.qname;  %address.content; >
<!-- end of address.element -->]]>

<!ENTITY % address.attlist  "INCLUDE" >
```

```
<![%address.attlist;[
<!ATTLIST %address.qname;
    %Common.attrib;
>
<!-- end of address.attlist -->]]>

<!ENTITY % blockquote.element  "INCLUDE" >
<![%blockquote.element;[
<!ENTITY % blockquote.content
   "( %Block.mix; )+"
>
<!ENTITY % blockquote.qname  "blockquote" >
<!ELEMENT %blockquote.qname;  %blockquote.content; >
<!-- end of blockquote.element -->]]>

<!ENTITY % blockquote.attlist  "INCLUDE" >
<![%blockquote.attlist;[
<!ATTLIST %blockquote.qname;
    %Common.attrib;
    cite        %URI.datatype;          #IMPLIED
>
<!-- end of blockquote.attlist -->]]>

<!ENTITY % pre.element  "INCLUDE" >
<![%pre.element;[
<!ENTITY % pre.content
   "( #PCDATA
    | %InlStruct.class;
    %InlPhras.class;
    | %tt.qname; | %i.qname; | %b.qname;
    %I18n.class;
    %Anchor.class;
    | %script.qname; | %map.qname;
    %Inline.extra; )*"
>
<!ENTITY % pre.qname  "pre" >
<!ELEMENT %pre.qname;  %pre.content; >
<!-- end of pre.element -->]]>

<!ENTITY % pre.attlist  "INCLUDE" >
<![%pre.attlist;[
<!ATTLIST %pre.qname;
    %Common.attrib;
    xml:space    (default|preserve)            #FIXED 'preserve'
>
<!-- end of pre.attlist -->]]>

<!-- .....................  Heading Elements  ..................... -->
```

```
<!ENTITY % Heading.content  "( #PCDATA | %Inline.mix; )*" >

<!ENTITY % h1.element  "INCLUDE" >
<![%h1.element;[
<!ENTITY % h1.qname  "h1" >
<!ELEMENT %h1.qname;  %Heading.content; >
<!-- end of h1.element -->]]>

<!ENTITY % h1.attlist  "INCLUDE" >
<![%h1.attlist;[
<!ATTLIST %h1.qname;
    %Common.attrib;
>
<!-- end of h1.attlist -->]]>

<!ENTITY % h2.element  "INCLUDE" >
<![%h2.element;[
<!ENTITY % h2.qname  "h2" >
<!ELEMENT %h2.qname;  %Heading.content; >
<!-- end of h2.element -->]]>

<!ENTITY % h2.attlist  "INCLUDE" >
<![%h2.attlist;[
<!ATTLIST %h2.qname;
    %Common.attrib;
>
<!-- end of h2.attlist -->]]>

<!ENTITY % h3.element  "INCLUDE" >
<![%h3.element;[
<!ENTITY % h3.qname  "h3" >
<!ELEMENT %h3.qname;  %Heading.content; >
<!-- end of h3.element -->]]>

<!ENTITY % h3.attlist  "INCLUDE" >
<![%h3.attlist;[
<!ATTLIST %h3.qname;
    %Common.attrib;
>
<!-- end of h3.attlist -->]]>

<!ENTITY % h4.element  "INCLUDE" >
<![%h4.element;[
<!ENTITY % h4.qname  "h4" >
<!ELEMENT %h4.qname;  %Heading.content; >
<!-- end of h4.element -->]]>
```

```
<!ENTITY % h4.attlist  "INCLUDE" >
<![%h4.attlist;[
<!ATTLIST %h4.qname;
    %Common.attrib;
>
<!-- end of h4.attlist -->]]>

<!ENTITY % h5.element  "INCLUDE" >
<![%h5.element;[
<!ENTITY % h5.qname  "h5" >
<!ELEMENT %h5.qname;  %Heading.content; >
<!-- end of h5.element -->]]>

<!ENTITY % h5.attlist  "INCLUDE" >
<![%h5.attlist;[
<!ATTLIST %h5.qname;
    %Common.attrib;
>
<!-- end of h5.attlist -->]]>

<!ENTITY % h6.element  "INCLUDE" >
<![%h6.element;[
<!ENTITY % h6.qname  "h6" >
<!ELEMENT %h6.qname;  %Heading.content; >
<!-- end of h6.element -->]]>

<!ENTITY % h6.attlist  "INCLUDE" >
<![%h6.attlist;[
<!ATTLIST %h6.qname;
    %Common.attrib;
>
<!-- end of h6.attlist -->]]>

<!-- end of xhtml-blkphras-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/rddl/xhtml-blkphras-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ....................................................................... -->
<!-- XHTML Legacy Markup Module ............................................ -->
<!-- file: xhtml-legacy-1.mod

   This is an extension of XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2005 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-legacy-1.mod,v 4.1 2001/04/10 09:42:30 altheim Exp $ SMI
```

```
    This DTD module is identified by the PUBLIC and SYSTEM identifiers:

      PUBLIC "-//W3C//ELEMENTS XHTML Legacy Markup 1.0//EN"
      SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-legacy-1.mod"

    Revisions:
    (none)
    ..... -->

<!-- HTML Legacy Markup

      font, basefont, center, s, strike, u, dir, menu, isindex

       (plus additional datatypes and attributes)

    This optional module declares additional markup for simple
    presentation-related markup based on features found in the
    HTML 4 Transitional and Frameset DTDs. This relies on
    inclusion of the Legacy Redeclarations module. This module
    also declares the frames, inline frames and object modules.

    This is to allow XHTML 1.1 documents to be transformed for
    display on HTML browsers where CSS support is inconsistent
    or unavailable.
-->
<!-- Constructing a Legacy DTD

    To construct a DTD driver obtaining a close approximation
    of the HTML 4 Transitional and Frameset DTDs, declare the
    Legacy Redeclarations module as the pre-framework redeclaration
    parameter entity (%xhtml-prefw-redecl.mod;) and INCLUDE its
    conditional section:

      ...
      <!ENTITY % xhtml-prefw-redecl.module "INCLUDE" >
      <![%xhtml-prefw-redecl.module;[
      <!ENTITY % xhtml-prefw-redecl.mod
        PUBLIC "-//W3C//ELEMENTS XHTML Legacy Redeclarations 1.0//EN"
            "xhtml-legacy-redecl-1.mod" >
      %xhtml-prefw-redecl.mod;]]>

    Such a DTD should be named with a variant FPI and redeclare
    the value of the %XHTML.version; parameter entity to that FPI:

      "-//Your Name Here//DTD XHTML Legacy 1.1//EN"

    IMPORTANT:  see also the notes included in the Legacy Redeclarations
    Module for information on how to construct a DTD using this module.
```

```
-->


<!-- Additional Element Types ..................................... -->

<!-- font: Local Font Modifier  ..................... -->

<!ENTITY % font.element  "INCLUDE" >
<![%font.element;[
<!ENTITY % font.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % font.qname  "font" >
<!ELEMENT %font.qname;  %font.content; >
<!-- end of font.element -->]]>

<!ENTITY % font.attlist  "INCLUDE" >
<![%font.attlist;[
<!ATTLIST %font.qname;
    %Core.attrib;
    %I18n.attrib;
    size        CDATA                #IMPLIED
    color       %Color.datatype;     #IMPLIED
    face        CDATA                #IMPLIED
>
<!-- end of font.attlist -->]]>

<!-- basefont: Base Font Size  ..................... -->

<!ENTITY % basefont.element  "INCLUDE" >
<![%basefont.element;[
<!ENTITY % basefont.content "EMPTY" >
<!ENTITY % basefont.qname  "basefont" >
<!ELEMENT %basefont.qname;  %basefont.content; >
<!-- end of basefont.element -->]]>

<!ENTITY % basefont.attlist  "INCLUDE" >
<![%basefont.attlist;[
<!ATTLIST %basefont.qname;
    %id.attrib;
    size        CDATA                #REQUIRED
    color       %Color.datatype;     #IMPLIED
    face        CDATA                #IMPLIED
>
<!-- end of basefont.attlist -->]]>

<!-- center: Center Alignment  ..................... -->
```

```
<!ENTITY % center.element  "INCLUDE" >
<![%center.element;[
<!ENTITY % center.content
   "( #PCDATA | %Flow.mix; )*"
>
<!ENTITY % center.qname  "center" >
<!ELEMENT %center.qname;  %center.content; >
<!-- end of center.element -->]]>

<!ENTITY % center.attlist  "INCLUDE" >
<![%center.attlist;[
<!ATTLIST %center.qname;
    %Common.attrib;
>
<!-- end of center.attlist -->]]>

<!-- s: Strike-Thru Text Style  ..... -->

<!ENTITY % s.element  "INCLUDE" >
<![%s.element;[
<!ENTITY % s.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % s.qname  "s" >
<!ELEMENT %s.qname;  %s.content; >
<!-- end of s.element -->]]>

<!ENTITY % s.attlist  "INCLUDE" >
<![%s.attlist;[
<!ATTLIST %s.qname;
    %Common.attrib;
>
<!-- end of s.attlist -->]]>

<!-- strike: Strike-Thru Text Style  .....-->

<!ENTITY % strike.element  "INCLUDE" >
<![%strike.element;[
<!ENTITY % strike.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % strike.qname  "strike" >
<!ELEMENT %strike.qname;  %strike.content; >
<!-- end of strike.element -->]]>

<!ENTITY % strike.attlist  "INCLUDE" >
<![%strike.attlist;[
<!ATTLIST %strike.qname;
```

```
    %Common.attrib;
>
<!-- end of strike.attlist -->]]>

<!-- u: Underline Text Style  ..........................-->

<!ENTITY % u.element  "INCLUDE" >
<![%u.element;[
<!ENTITY % u.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ENTITY % u.qname  "u" >
<!ELEMENT %u.qname;  %u.content; >
<!-- end of u.element -->]]>

<!ENTITY % u.attlist  "INCLUDE" >
<![%u.attlist;[
<!ATTLIST %u.qname;
    %Common.attrib;
>
<!-- end of u.attlist -->]]>

<!-- dir: Directory List  ..... -->

<!-- NOTE: the content model for <dir> in HTML 4 excluded %Block.mix;
-->
<!ENTITY % dir.element  "INCLUDE" >
<![%dir.element;[
<!ENTITY % dir.content
   "( %li.qname; )+"
>
<!ENTITY % dir.qname  "dir" >
<!ELEMENT %dir.qname;  %dir.content; >
<!-- end of dir.element -->]]>

<!ENTITY % dir.attlist  "INCLUDE" >
<![%dir.attlist;[
<!ATTLIST %dir.qname;
    %Common.attrib;
    compact      ( compact )            #IMPLIED
>
<!-- end of dir.attlist -->]]>

<!-- menu: Menu List  ..... -->

<!-- NOTE: the content model for <menu> in HTML 4 excluded %Block.mix;
-->
<!ENTITY % menu.element  "INCLUDE" >
```

```
<![%menu.element;[
<!ENTITY % menu.content
   "( %li.qname; )+"
>
<!ENTITY % menu.qname  "menu" >
<!ELEMENT %menu.qname;  %menu.content; >
<!-- end of menu.element -->]]>

<!ENTITY % menu.attlist  "INCLUDE" >
<![%menu.attlist;[
<!ATTLIST %menu.qname;
    %Common.attrib;
    compact      ( compact )             #IMPLIED
>
<!-- end of menu.attlist -->]]>

<!-- isindex: Single-Line Prompt  ..... -->

<!ENTITY % isindex.element  "INCLUDE" >
<![%isindex.element;[
<!ENTITY % isindex.content "EMPTY" >
<!ENTITY % isindex.qname  "isindex" >
<!ELEMENT %isindex.qname;  %isindex.content; >
<!-- end of isindex.element -->]]>

<!ENTITY % isindex.attlist  "INCLUDE" >
<![%isindex.attlist;[
<!ATTLIST %isindex.qname;
    %Core.attrib;
    %I18n.attrib;
    prompt       %Text.datatype;          #IMPLIED
>
<!-- end of isindex.attlist -->]]>


<!-- Additional Attributes ..... -->

<!-- Alignment attribute for Transitional use in HTML browsers
   (this functionality is generally well-supported in CSS,
   except within some contexts)
-->
<!ENTITY % align.attrib
   "align       ( left | center | right | justify ) #IMPLIED"
>

<!ATTLIST %applet.qname;
    align       ( top | middle | bottom | left | right ) #IMPLIED
    hspace      %Pixels.datatype;         #IMPLIED
```

```
    vspace      %Pixels.datatype;        #IMPLIED
>

<!ATTLIST %body.qname;
    background   %URI.datatype;          #IMPLIED
    bgcolor      %Color.datatype;        #IMPLIED
    text         %Color.datatype;        #IMPLIED
    link         %Color.datatype;        #IMPLIED
    vlink        %Color.datatype;        #IMPLIED
    alink        %Color.datatype;        #IMPLIED
>

<!ATTLIST %br.qname;
    clear        ( left | all | right | none ) 'none'
>

<!ATTLIST %caption.qname;
    align        ( top | bottom | left | right ) #IMPLIED
>

<!ATTLIST %div.qname;
    %align.attrib;
>

<!ATTLIST %h1.qname;
    %align.attrib;
>

<!ATTLIST %h2.qname;
    %align.attrib;
>

<!ATTLIST %h3.qname;
    %align.attrib;
>

<!ATTLIST %h4.qname;
    %align.attrib;
>

<!ATTLIST %h5.qname;
    %align.attrib;
>

<!ATTLIST %h6.qname;
    %align.attrib;
>
```

```
<!ATTLIST %hr.qname;
    align       ( left | center | right ) #IMPLIED
    noshade     ( noshade )             #IMPLIED
    size        %Pixels.datatype;       #IMPLIED
    width       %Length.datatype;       #IMPLIED
>

<!ATTLIST %img.qname;
    align      ( top | middle | bottom | left | right ) #IMPLIED
    border     %Pixels.datatype;        #IMPLIED
    hspace     %Pixels.datatype;        #IMPLIED
    vspace     %Pixels.datatype;        #IMPLIED
>

<!ATTLIST %input.qname;
    align      ( top | middle | bottom | left | right ) #IMPLIED
>

<!ATTLIST %legend.qname;
    align       ( top | bottom | left | right ) #IMPLIED
>

<!ATTLIST %li.qname;
    type        CDATA                   #IMPLIED
    value       %Number.datatype;       #IMPLIED
>

<!ATTLIST %object.qname;
    align       ( top | middle | bottom | left | right ) #IMPLIED
    border      %Pixels.datatype;       #IMPLIED
    hspace      %Pixels.datatype;       #IMPLIED
    vspace      %Pixels.datatype;       #IMPLIED
>

<!ATTLIST %dl.qname;
    compact     ( compact )             #IMPLIED
>

<!ATTLIST %ol.qname;
    type        CDATA                   #IMPLIED
    compact     ( compact )             #IMPLIED
    start       %Number.datatype;       #IMPLIED
>

<!ATTLIST %p.qname;
    %align.attrib;
>
```

```
<!ATTLIST %pre.qname;
    width       %Length.datatype;       #IMPLIED
>

<!ATTLIST %script.qname;
    language    %ContentType.datatype;  #IMPLIED
>

<!ATTLIST %table.qname;
    align       ( left | center | right ) #IMPLIED
    bgcolor     %Color.datatype;        #IMPLIED
>

<!ATTLIST %tr.qname;
    bgcolor     %Color.datatype;        #IMPLIED
>

<!ATTLIST %th.qname;
    nowrap      ( nowrap )              #IMPLIED
    bgcolor     %Color.datatype;        #IMPLIED
    width       %Length.datatype;       #IMPLIED
    height      %Length.datatype;       #IMPLIED
>

<!ATTLIST %td.qname;
    nowrap      ( nowrap )              #IMPLIED
    bgcolor     %Color.datatype;        #IMPLIED
    width       %Length.datatype;       #IMPLIED
    height      %Length.datatype;       #IMPLIED
>

<!ATTLIST %ul.qname;
    type        CDATA                   #IMPLIED
    compact     ( compact )             #IMPLIED
>

<!-- Frames Module ..... -->
<!ENTITY % xhtml-frames.module "IGNORE" >
<![%xhtml-frames.module;[
<!ENTITY % xhtml-frames.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Frames 1.0//EN"
       "xhtml-frames-1.mod" >
%xhtml-frames.mod;]]>

<!-- Inline Frames Module ..... -->
<!ENTITY % xhtml-iframe.module "INCLUDE" >
<![%xhtml-iframe.module;[
<!ATTLIST %iframe.qname;
```

```
    align       ( top | middle | bottom | left | right ) #IMPLIED
>
<!ENTITY % xhtml-iframe.mod
   PUBLIC "-//W3C//ELEMENTS XHTML Inline Frame Element 1.0//EN"
         "xhtml-iframe-1.mod" >
%xhtml-iframe.mod;]]>

<!-- end of xhtml-legacy-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-legacy-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Hypertext Attributes Module  ..................................... -->
<!-- file: xhtml-hyperAttributes-1.mod

   This is XHTML-RDFa, modules to annotate XHTML family documents.
   Copyright 2007 W3C (MIT, ERCIM, Keio), All Rights Reserved.
   Revision: $Id: xhtml-hyperAttributes-1.mod,v 1.2 2007/08/11 19:06:51 jigsaw Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES XHTML HyperAttributes 1.0//EN"
     SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-hyperAttributes-1.mod"

   Revisions:
   ..................................................................... -->

<!-- Common Attributes

   This module declares a collection of hypertext related
   attributes.

   %NS.decl.attrib; is declared in the XHTML Qname module.

  This file also includes declarations of "global" versions of the
   attributes.  The global versions of the attributes are for use on
   elements in other namespaces.
-->

<!ENTITY % href.attrib
   "href       %URI.datatype;           #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.href.attrib
```

```
   "%XHTML.prefix;:href          %URI.datatype;       #IMPLIED"
>
]]>

<!ENTITY % hreflang.attrib
   "hreflang       %LanguageCode.datatype;            #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.hreflang.attrib
   "%XHTML.prefix;:hreflang          %LanguageCode.datatype;       #IMPLIED"
>
]]>

<!ENTITY % hrefmedia.attrib
   "hrefmedia       %MediaDesc.datatype;            #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.hrefmedia.attrib
   "%XHTML.prefix;:hrefmedia          %MediaDesc.datatype;       #IMPLIED"
>
]]>

<!ENTITY % hreftype.attrib
   "hreftype       %ContentTypes.datatype;            #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.hreftype.attrib
   "%XHTML.prefix;:hreftype          %ContentTypes.datatype;       #IMPLIED"
>
]]>

<!ENTITY % Hyper.attrib.extra "" >

<!ENTITY % Hyper.attrib
   "%href.attrib;
    %hreflang.attrib;
    %hrefmedia.attrib;
    %hreftype.attrib;
    %Hyper.attrib.extra;"
>

<!ENTITY % XHTML.global.hyper.attrib.extra "" >

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.hyper.attrib
```

```
   "%XHTML.global.href.attrib;
    %XHTML.global.hreflang.attrib;
    %XHTML.global.hrefmedia.attrib;
    %XHTML.global.hreftype.attrib;
    %XHTML.global.hyper.attrib.extra;"
>
]]>

<!ENTITY % XHTML.global.hyper.attrib "" >

<!-- end of xhtml-hyperAttributes-1.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/xhtml11/xhtml-hyperAttributes-1.mod
No license file was found, but licenses were detected in source scan.

```
<!-- ..................................................................... -->
<!-- XHTML Link Element Module  ..... -->
<!-- file: xhtml-link-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-link-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Link Element 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-link-1.mod"

   Revisions:
   (none)
   ..... -->

<!-- Link element

      link

   This module declares the link element type and its attributes,
   which could (in principle) be used to define document-level links
   to external resources such as:

   a) for document specific toolbars/menus, e.g. start, contents,
      previous, next, index, end, help
   b) to link to a separate style sheet (rel="stylesheet")
   c) to make a link to a script (rel="script")
   d) by stylesheets to control how collections of html nodes are
      rendered into printed documents
```

```
   e) to make a link to a printable version of this document
      e.g. a postscript or pdf version (rel="alternate" media="print")
-->

<!-- link: Media-Independent Link ..... -->

<!ENTITY % link.element  "INCLUDE" >
<![%link.element;[
<!ENTITY % link.content  "EMPTY" >
<!ENTITY % link.qname  "link" >
<!ELEMENT %link.qname;  %link.content; >
<!-- end of link.element -->]]>

<!ENTITY % link.attlist  "INCLUDE" >
<![%link.attlist;[
<!ATTLIST %link.qname;
    %Common.attrib;
    charset      %Charset.datatype;       #IMPLIED
    href         %URI.datatype;           #IMPLIED
    hreflang     %LanguageCode.datatype;  #IMPLIED
    type         %ContentType.datatype;   #IMPLIED
    rel          %LinkTypes.datatype;     #IMPLIED
    rev          %LinkTypes.datatype;     #IMPLIED
    media        %MediaDesc.datatype;     #IMPLIED
>
<!-- end of link.attlist -->]]>

<!-- end of xhtml-link-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-link-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- XHTML Lists Module  ..... -->
<!-- file: xhtml-list-1.mod

   This is XHTML, a reformulation of HTML as a modular XML application.
   Copyright 1998-2000 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: xhtml-list-1.mod,v 1.1 2001/02/13 12:24:22 ht Exp $ SMI

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ELEMENTS XHTML Lists 1.0//EN"
     SYSTEM "http://www.w3.org/TR/xhtml-modulatization/DTD/xhtml-list-1.mod"

   Revisions:
```

```
   (none)
   ..... -->

<!-- Lists

      dl, dt, dd, ol, ul, li

   This module declares the list-oriented element types
   and their attributes.
-->

<!ENTITY % dl.qname  "dl" >
<!ENTITY % dt.qname  "dt" >
<!ENTITY % dd.qname  "dd" >
<!ENTITY % ol.qname  "ol" >
<!ENTITY % ul.qname  "ul" >
<!ENTITY % li.qname  "li" >

<!-- dl: Definition List ..... -->

<!ENTITY % dl.element  "INCLUDE" >
<![%dl.element;[
<!ENTITY % dl.content  "( %dt.qname; | %dd.qname; )+" >
<!ELEMENT %dl.qname;  %dl.content; >
<!-- end of dl.element -->]]>

<!ENTITY % dl.attlist  "INCLUDE" >
<![%dl.attlist;[
<!ATTLIST %dl.qname;
    %Common.attrib;
>
<!-- end of dl.attlist -->]]>

<!-- dt: Definition Term ..... -->

<!ENTITY % dt.element  "INCLUDE" >
<![%dt.element;[
<!ENTITY % dt.content
   "( #PCDATA | %Inline.mix; )*"
>
<!ELEMENT %dt.qname;  %dt.content; >
<!-- end of dt.element -->]]>

<!ENTITY % dt.attlist  "INCLUDE" >
<![%dt.attlist;[
<!ATTLIST %dt.qname;
    %Common.attrib;
>
```

```
<!-- end of dt.attlist -->]]>

<!-- dd: Definition Description ..... -->

<!ENTITY % dd.element  "INCLUDE" >
<![%dd.element;[
<!ENTITY % dd.content
   "( #PCDATA | %Flow.mix; )*"
>
<!ELEMENT %dd.qname;  %dd.content; >
<!-- end of dd.element -->]]>

<!ENTITY % dd.attlist  "INCLUDE" >
<![%dd.attlist;[
<!ATTLIST %dd.qname;
    %Common.attrib;
>
<!-- end of dd.attlist -->]]>

<!-- ol: Ordered List (numbered styles) ..... -->

<!ENTITY % ol.element  "INCLUDE" >
<![%ol.element;[
<!ENTITY % ol.content  "( %li.qname; )+" >
<!ELEMENT %ol.qname;  %ol.content; >
<!-- end of ol.element -->]]>

<!ENTITY % ol.attlist  "INCLUDE" >
<![%ol.attlist;[
<!ATTLIST %ol.qname;
    %Common.attrib;
>
<!-- end of ol.attlist -->]]>

<!-- ul: Unordered List (bullet styles) ..... -->

<!ENTITY % ul.element  "INCLUDE" >
<![%ul.element;[
<!ENTITY % ul.content  "( %li.qname; )+" >
<!ELEMENT %ul.qname;  %ul.content; >
<!-- end of ul.element -->]]>

<!ENTITY % ul.attlist  "INCLUDE" >
<![%ul.attlist;[
<!ATTLIST %ul.qname;
    %Common.attrib;
>
<!-- end of ul.attlist -->]]>
```

```
<!-- li: List Item ..... -->

<!ENTITY % li.element  "INCLUDE" >
<![%li.element;[
<!ENTITY % li.content
   "( #PCDATA | %Flow.mix; )*"
>
<!ELEMENT %li.qname;  %li.content; >
<!-- end of li.element -->]]>

<!ENTITY % li.attlist  "INCLUDE" >
<![%li.attlist;[
<!ATTLIST %li.qname;
    %Common.attrib;
>
<!-- end of li.attlist -->]]>

<!-- end of xhtml-list-1.mod -->

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-
jar/net/sf/saxon/data/w3c/rddl/xhtml-list-1.mod
No license file was found, but licenses were detected in source scan.

<!-- ..... -->
<!-- SVG 1.1 Viewport Attribute Module ..... -->
<!-- file: svg-viewport-attrib.mod

   This is SVG, a language for describing two-dimensional graphics in XML.
   Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
   Revision: $Id: svg-viewport-attrib.mod,v 1.2 2002/04/20 18:07:43 fujisawa Exp $

   This DTD module is identified by the PUBLIC and SYSTEM identifiers:

     PUBLIC "-//W3C//ENTITIES SVG 1.1 Viewport Attribute//EN"
     SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-viewport-attrib.mod"

   ..... -->

<!-- Viewport Attribute

     clip, overflow

   This module defines the Viewport attribute set.
-->

<!-- 'clip' property/attribute value (e.g., 'auto', rect(...)) -->
```

```
<!ENTITY % ClipValue.datatype "CDATA" >

<!ENTITY % SVG.clip.attrib
  "clip %ClipValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.overflow.attrib
  "overflow ( visible | hidden | scroll | auto | inherit ) #IMPLIED"
>

<!ENTITY % SVG.Viewport.extra.attrib "" >

<!ENTITY % SVG.Viewport.attrib
  "%SVG.clip.attrib;
   %SVG.overflow.attrib;
   %SVG.Viewport.extra.attrib;"
>

<!-- end of svg-viewport-attrib.mod -->
```

Found in path(s):
* /opt/cola/permits/1340815750_1654855500.4515722/0/saxon-he-10-6-sources-jar/net/sf/saxon/data/w3c/svg11/svg-viewport-attrib.mod

1.70 brave-instrumentation-http-adapters 5.13.9

1.70.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the

direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and

wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor

has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0
* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed
by Square, Inc in the Moshi Library:

* Copyright (C) 2010 Google Inc.
* License: Apache License v2.0
* Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

* License: Apache License v2.0
* Homepage: https://github.com/raphw/weak-lock-free

1.71 hikaricp HikariCP-4.0.3

1.71.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright
owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities
that control, are controlled by, or are under common control with that entity.
For the purposes of this definition, "control" means (i) the power, direct or
indirect, to cause the direction or management of such entity, whether by
contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising
permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including
but not limited to software source code, documentation source, and configuration
files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor

that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding

such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate

notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.72 brave-context:-slf4j 5.13.9

1.72.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity

exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0
* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed

by Square, Inc in the Moshi Library:

* Copyright (C) 2010 Google Inc.
* License: Apache License v2.0
* Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

* License: Apache License v2.0
* Homepage: https://github.com/raphw/weak-lock-free

1.73 zipkin-reporter-metrics-by-micrometer 2.16.3

1.73.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.74 spring-cloud-commons 3.1.3

1.74.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the

Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.75 okhttp-logging-interceptor 4.9.3

1.75.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2015 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1288520085_1648141986.97/0/logging-interceptor-4-9-3-sources-jar/okhttp3/logging/utf8.kt
* /opt/cola/permits/1288520085_1648141986.97/0/logging-interceptor-4-9-3-sources-jar/okhttp3/logging/HttpLoggingInterceptor.kt

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1288520085_1648141986.97/0/logging-interceptor-4-9-3-sources-jar/okhttp3/logging/LoggingEventListener.kt

1.76 reflections 0.9.12

1.76.1 Available under license :

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
Version 2, December 2004

Copyright (C) 2004 Sam Hocevar <sam@hocevar.net>

Everyone is permitted to copy and distribute verbatim or modified
copies of this license document, and changing it is allowed as long
as the name is changed.

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. You just DO WHAT THE FUCK YOU WANT TO.

1.77 mockwebserver 4.10.0

1.77.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2012 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137136_1671710651.7084372/0/mockwebserver-4-10-0-sources-jar/okhttp3/mockwebserver/Dispatcher.kt
* /opt/cola/permits/1519137136_1671710651.7084372/0/mockwebserver-4-10-0-sources-jar/okhttp3/mockwebserver/QueueDispatcher.kt
No license file was found, but licenses were detected in source scan.

/*

```
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519137136_1671710651.7084372/0/mockwebserver-4-10-0-sources-jar/okhttp3/mockwebserver/internal/duplex/DuplexResponseBody.kt
* /opt/cola/permits/1519137136_1671710651.7084372/0/mockwebserver-4-10-0-sources-jar/okhttp3/internal/duplex/MwsDuplexAccess.kt
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519137136_1671710651.7084372/0/mockwebserver-4-10-0-sources-jar/okhttp3/mockwebserver/internal/duplex/MockDuplexResponseBody.kt
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2014 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
```

* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137136_1671710651.7084372/0/mockwebserver-4-10-0-sources-jar/okhttp3/mockwebserver/PushPromise.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137136_1671710651.7084372/0/mockwebserver-4-10-0-sources-jar/okhttp3/mockwebserver/SocketPolicy.kt
* /opt/cola/permits/1519137136_1671710651.7084372/0/mockwebserver-4-10-0-sources-jar/okhttp3/mockwebserver/MockResponse.kt
* /opt/cola/permits/1519137136_1671710651.7084372/0/mockwebserver-4-10-0-sources-jar/okhttp3/mockwebserver/RecordedRequest.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 Google Inc.
* Copyright (C) 2013 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137136_1671710651.7084372/0/mockwebserver-4-10-0-sources-jar/okhttp3/mockwebserver/MockWebServer.kt

1.78 fabric8-::-kubernetes-model-::-rbac 5.12.1

1.78.1 Available under license :

No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340816216_1654860735.5882082/0/kubernetes-model-rbac-5-12-1-sources-jar/manifest.vm
No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.

You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340816216_1654860735.5882082/0/kubernetes-model-rbac-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-rbac/pom.xml

1.79 brave-instrumentation-jms 5.13.9

1.79.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0
* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed
by Square, Inc in the Moshi Library:

* Copyright (C) 2010 Google Inc.
* License: Apache License v2.0
* Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

 * License: Apache License v2.0
 * Homepage: https://github.com/raphw/weak-lock-free

1.80 jna 5.8.0

1.80.1 Available under license :

/*
* The contents of this file are subject to the Mozilla Public License Version 1.1
* (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at http://www.mozilla.org/MPL/
*
* Software distributed under the License is distributed on an "AS IS" basis,
* WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
* for the specific language governing rights and limitations under the License.
*
* The Original Code is 'iText, a free JAVA-PDF library'.
*
* The Initial Developer of the Original Code is Bruno Lowagie. Portions created by
* the Initial Developer are Copyright (C) 1999, 2000, 2001, 2002 by Bruno Lowagie.
* All Rights Reserved.
* Co-Developer of the code is Paulo Soares. Portions created by the Co-Developer
* are Copyright (C) 2000, 2001, 2002 by Paulo Soares. All Rights Reserved.
*
* Contributor(s): all the names of the contributors are added in the source code
* where applicable.
*
* Alternatively, the contents of this file may be used under the terms of the
* LGPL license (the "GNU LIBRARY GENERAL PUBLIC LICENSE"), in which case the
* provisions of LGPL are applicable instead of those above. If you wish to
* allow use of your version of this file only under the terms of the LGPL
* License and not to allow others to use your version of this file under
* the MPL, indicate your decision by deleting the provisions above and
* replace them with the notice and other provisions required by the LGPL.
* If you do not delete the provisions above, a recipient may use your version
* of this file under either the MPL or the GNU LIBRARY GENERAL PUBLIC LICENSE.
*
* This library is free software; you can redistribute it and/or modify it
* under the terms of the MPL as stated above or under the terms of the GNU
* Library General Public License as published by the Free Software Foundation;
* either version 2 of the License, or any later version.
*
* This library is distributed in the hope that it will be useful, but WITHOUT
* ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS

* FOR A PARTICULAR PURPOSE. See the GNU Library general Public License for more
* details.
*
* If you didn't download this code from the following link, you should check if
* you aren't using an obsolete version:
* http://www.lowagie.com/iText/
*
* This class is generated based on a grammar file privided by SUN, and updated
* by Carsten Hammer. SUN's license agreement can be found at this URL:
* http://java.sun.com/products/java-media/2D/samples/samples-license.html
* See also the file sun.txt in directory com.lowagie.text.pdf
*/

Maven Ant Tasks
Copyright 2002-2011 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
==========
==========
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from 'org.jfree' to 'clover.org.jfree'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.
==========
==========

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser

General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and

distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public

License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data

structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials

specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE

LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!
BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.
Apache Ant
Copyright 1999-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The <sync> task is based on code Copyright (c) 2002, Landmark
Graphics Corp that has been kindly donated to the Apache Software
Foundation.
/*
* Apache License
* Version 2.0, January 2004
* http://www.apache.org/licenses/

*
* TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
*
* 1. Definitions.
*
*   "License" shall mean the terms and conditions for use, reproduction,
*   and distribution as defined by Sections 1 through 9 of this document.
*
*   "Licensor" shall mean the copyright owner or entity authorized by
*   the copyright owner that is granting the License.
*
*   "Legal Entity" shall mean the union of the acting entity and all
*   other entities that control, are controlled by, or are under common
*   control with that entity. For the purposes of this definition,
*   "control" means (i) the power, direct or indirect, to cause the
*   direction or management of such entity, whether by contract or
*   otherwise, or (ii) ownership of fifty percent (50%) or more of the
*   outstanding shares, or (iii) beneficial ownership of such entity.
*
*   "You" (or "Your") shall mean an individual or Legal Entity
*   exercising permissions granted by this License.
*
*   "Source" form shall mean the preferred form for making modifications,
*   including but not limited to software source code, documentation
*   source, and configuration files.
*
*   "Object" form shall mean any form resulting from mechanical
*   transformation or translation of a Source form, including but
*   not limited to compiled object code, generated documentation,
*   and conversions to other media types.
*
*   "Work" shall mean the work of authorship, whether in Source or
*   Object form, made available under the License, as indicated by a
*   copyright notice that is included in or attached to the work
*   (an example is provided in the Appendix below).
*
*   "Derivative Works" shall mean any work, whether in Source or Object
*   form, that is based on (or derived from) the Work and for which the
*   editorial revisions, annotations, elaborations, or other modifications
*   represent, as a whole, an original work of authorship. For the purposes
*   of this License, Derivative Works shall not include works that remain
*   separable from, or merely link (or bind by name) to the interfaces of,
*   the Work and Derivative Works thereof.
*
*   "Contribution" shall mean any work of authorship, including
*   the original version of the Work and any modifications or additions
*   to that Work or Derivative Works thereof, that is intentionally
*   submitted to Licensor for inclusion in the Work by the copyright owner

```
*      or by an individual or Legal Entity authorized to submit on behalf of
*      the copyright owner. For the purposes of this definition, "submitted"
*      means any form of electronic, verbal, or written communication sent
*      to the Licensor or its representatives, including but not limited to
*      communication on electronic mailing lists, source code control systems,
*      and issue tracking systems that are managed by, or on behalf of, the
*      Licensor for the purpose of discussing and improving the Work, but
*      excluding communication that is conspicuously marked or otherwise
*      designated in writing by the copyright owner as "Not a Contribution."
*
*      "Contributor" shall mean Licensor and any individual or Legal Entity
*      on behalf of whom a Contribution has been received by Licensor and
*      subsequently incorporated within the Work.
*
*   2. Grant of Copyright License. Subject to the terms and conditions of
*      this License, each Contributor hereby grants to You a perpetual,
*      worldwide, non-exclusive, no-charge, royalty-free, irrevocable
*      copyright license to reproduce, prepare Derivative Works of,
*      publicly display, publicly perform, sublicense, and distribute the
*      Work and such Derivative Works in Source or Object form.
*
*   3. Grant of Patent License. Subject to the terms and conditions of
*      this License, each Contributor hereby grants to You a perpetual,
*      worldwide, non-exclusive, no-charge, royalty-free, irrevocable
*      (except as stated in this section) patent license to make, have made,
*      use, offer to sell, sell, import, and otherwise transfer the Work,
*      where such license applies only to those patent claims licensable
*      by such Contributor that are necessarily infringed by their
*      Contribution(s) alone or by combination of their Contribution(s)
*      with the Work to which such Contribution(s) was submitted. If You
*      institute patent litigation against any entity (including a
*      cross-claim or counterclaim in a lawsuit) alleging that the Work
*      or a Contribution incorporated within the Work constitutes direct
*      or contributory patent infringement, then any patent licenses
*      granted to You under this License for that Work shall terminate
*      as of the date such litigation is filed.
*
*   4. Redistribution. You may reproduce and distribute copies of the
*      Work or Derivative Works thereof in any medium, with or without
*      modifications, and in Source or Object form, provided that You
*      meet the following conditions:
*
*      (a) You must give any other recipients of the Work or
*          Derivative Works a copy of this License; and
*
*      (b) You must cause any modified files to carry prominent notices
*          stating that You changed the files; and
*
```

```
*      (c) You must retain, in the Source form of any Derivative Works
*          that You distribute, all copyright, patent, trademark, and
*          attribution notices from the Source form of the Work,
*          excluding those notices that do not pertain to any part of
*          the Derivative Works; and
*
*      (d) If the Work includes a "NOTICE" text file as part of its
*          distribution, then any Derivative Works that You distribute must
*          include a readable copy of the attribution notices contained
*          within such NOTICE file, excluding those notices that do not
*          pertain to any part of the Derivative Works, in at least one
*          of the following places: within a NOTICE text file distributed
*          as part of the Derivative Works; within the Source form or
*          documentation, if provided along with the Derivative Works; or,
*          within a display generated by the Derivative Works, if and
*          wherever such third-party notices normally appear. The contents
*          of the NOTICE file are for informational purposes only and
*          do not modify the License. You may add Your own attribution
*          notices within Derivative Works that You distribute, alongside
*          or as an addendum to the NOTICE text from the Work, provided
*          that such additional attribution notices cannot be construed
*          as modifying the License.
*
*      You may add Your own copyright statement to Your modifications and
*      may provide additional or different license terms and conditions
*      for use, reproduction, or distribution of Your modifications, or
*      for any such Derivative Works as a whole, provided Your use,
*      reproduction, and distribution of the Work otherwise complies with
*      the conditions stated in this License.
*
*   5. Submission of Contributions. Unless You explicitly state otherwise,
*      any Contribution intentionally submitted for inclusion in the Work
*      by You to the Licensor shall be under the terms and conditions of
*      this License, without any additional terms or conditions.
*      Notwithstanding the above, nothing herein shall supersede or modify
*      the terms of any separate license agreement you may have executed
*      with Licensor regarding such Contributions.
*
*   6. Trademarks. This License does not grant permission to use the trade
*      names, trademarks, service marks, or product names of the Licensor,
*      except as required for reasonable and customary use in describing the
*      origin of the Work and reproducing the content of the NOTICE file.
*
*   7. Disclaimer of Warranty. Unless required by applicable law or
*      agreed to in writing, Licensor provides the Work (and each
*      Contributor provides its Contributions) on an "AS IS" BASIS,
*      WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
*      implied, including, without limitation, any warranties or conditions
```

```
*      of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
*      PARTICULAR PURPOSE. You are solely responsible for determining the
*      appropriateness of using or redistributing the Work and assume any
*      risks associated with Your exercise of permissions under this License.
*
*   8. Limitation of Liability. In no event and under no legal theory,
*      whether in tort (including negligence), contract, or otherwise,
*      unless required by applicable law (such as deliberate and grossly
*      negligent acts) or agreed to in writing, shall any Contributor be
*      liable to You for damages, including any direct, indirect, special,
*      incidental, or consequential damages of any character arising as a
*      result of this License or out of the use or inability to use the
*      Work (including but not limited to damages for loss of goodwill,
*      work stoppage, computer failure or malfunction, or any and all
*      other commercial damages or losses), even if such Contributor
*      has been advised of the possibility of such damages.
*
*   9. Accepting Warranty or Additional Liability. While redistributing
*      the Work or Derivative Works thereof, You may choose to offer,
*      and charge a fee for, acceptance of support, warranty, indemnity,
*      or other liability obligations and/or rights consistent with this
*      License. However, in accepting such obligations, You may act only
*      on Your own behalf and on Your sole responsibility, not on behalf
*      of any other Contributor, and only if You agree to indemnify,
*      defend, and hold each Contributor harmless for any liability
*      incurred by, or claims asserted against, such Contributor by reason
*      of your accepting any such warranty or additional liability.
*
*   END OF TERMS AND CONDITIONS
*
*   APPENDIX: How to apply the Apache License to your work.
*
*      To apply the Apache License to your work, attach the following
*      boilerplate notice, with the fields enclosed by brackets "[]"
*      replaced with your own identifying information. (Don't include
*      the brackets!)  The text should be enclosed in the appropriate
*      comment syntax for the file format. We also recommend that a
*      file or class name and description of purpose be included on the
*      same "printed page" as the copyright notice for easier
*      identification within third-party archives.
*
*   Copyright [yyyy] [name of copyright owner]
*
*   Licensed under the Apache License, Version 2.0 (the "License");
*   you may not use this file except in compliance with the License.
*   You may obtain a copy of the License at
*
*       http://www.apache.org/licenses/LICENSE-2.0
```

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

W3C SOFTWARE NOTICE AND LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

This license came from: http://www.megginson.com/SAX/copying.html
However please note future versions of SAX may be covered
under http://saxproject.org/?selected=pd

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05
==========
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from 'com.keypoint/org.jfree' to 'clover.com.keypoint/clover.org.jfree'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.
==========

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts

as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is

linked with the Library has the freedom and the wherewithal to run
that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and
modification follow. Pay close attention to the difference between a
"work based on the library" and a "work that uses the library". The
former contains code derived from the library, whereas the latter must
be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other
program which contains a notice placed by the copyright holder or
other authorized party saying it may be distributed under the terms of
this Lesser General Public License (also called "this License").
Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data
prepared so as to be conveniently linked with application programs
(which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work
which has been distributed under these terms. A "work based on the
Library" means either the Library or any derivative work under
copyright law: that is to say, a work containing the Library or a
portion of it, either verbatim or with modifications and/or translated
straightforwardly into another language. (Hereinafter, translation is
included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for
making modifications to it. For a library, complete source code means
all the source code for all modules it contains, plus any associated
interface definition files, plus the scripts used to control compilation
and installation of the library.

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running a program using the Library is not restricted, and output from
such a program is covered only if its contents constitute a work based
on the Library (independent of the use of the Library in a tool for
writing it). Whether that is true depends on what the Library does
and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's
complete source code as you receive it, in any medium, provided that
you conspicuously and appropriately publish on each copy an
appropriate copyright notice and disclaimer of warranty; keep intact

all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest

your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License.

Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact

that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the

integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE

IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or

Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work

or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work

by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache Commons Lang
Copyright 2001-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

```
/*
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000-2003 The Apache Software Foundation.  All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. The end-user documentation included with the redistribution, if
*    any, must include the following acknowlegement:
*       "This product includes software developed by the
```

*   Apache Software Foundation (http://www.apache.org/)."
*   Alternately, this acknowlegement may appear in the software itself,
*   if and wherever such third-party acknowlegements normally appear.
*
* 4. The names "Ant" and "Apache Software
*   Foundation" must not be used to endorse or promote products derived
*   from this software without prior written permission. For written
*   permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
*   nor may "Apache" appear in their names without prior written
*   permission of the Apache Group.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ==
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.
*/
    GNU LESSER GENERAL PUBLIC LICENSE
                Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

 This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

 0. Additional Definitions.

 As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU

General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version

of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

MOZILLA PUBLIC LICENSE
Version 1.1

---

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous

Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.
The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or

as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.
Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims

infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.
The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.
Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.
You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters
(a) Third Party Claims.
If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the

party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.
If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.
Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.
You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.
You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the

obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.
You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.
Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.
Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.
If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declatory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom

You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN

INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is _____.

The Initial Developer of the Original Code is _____. Portions created by _____ are Copyright (C) _____ _____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[___] License"), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [___] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [___] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

/*

Copyright (c) 2000, Derek Petillo
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are

met:

Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

Neither the name of Praxis Software nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Public Domain Dedication
This license is acceptable for Free Cultural Works.

Copyright-Only Dedication (based on United States law) or Public Domain Certification

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed,

transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.
Open Source License for the overlibmws Package

1. License coverage

Note that this license only covers the script library (javascript core and plugin modules) and not any supporting material such as the overlibmws website or its online documentation and support files. You may not reproduce the website or its online material without explicit written permission from the author, but can freely incorporate scripts and procedures which are demonstrated in that material into your own HTML or XML documents.

2. License (Artistic)

Preamble

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:

"Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)

"Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.

You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:

place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as ftp.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

use the modified Package only within your corporation or organization.

rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

make other distribution arrangements with the Copyright Holder.

You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

accompany the distribution with the machine-readable source of the Package with your modifications.

accompany any non-standard executables with their corresponding Standard Version executables, giving the non-standard executables non-standard names, and clearly documenting the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

make other distribution arrangements with the Copyright Holder.

You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own.

The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whomever generated them, and may be sold commercially, and may be aggregated with this Package.

C or perl subroutines supplied by you and linked into this Package shall not be considered part of this Package.

The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

==========

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from 'it.unimi.dsi.fastutil' to 'clover.it.unimi.dsi.fastutil'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

==========

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using

a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License").

Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices

stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in

these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the
source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not
compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the
Library, but is designed to work with the Library by being compiled or
linked with it, is called a "work that uses the Library". Such a
work, in isolation, is not a derivative work of the Library, and
therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library
creates an executable that is a derivative of the Library (because it
contains portions of the Library), rather than a "work that uses the
library". The executable is therefore covered by this License.
Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file
that is part of the Library, the object code for the work may be a
derivative work of the Library even though the source code is not.
Whether this is true is especially significant if the work can be
linked without the Library, or if the work is itself a library. The
threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data
structure layouts and accessors, and small macros and small inline
functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative
work. (Executables containing this object code plus portions of the
Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the

Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus

excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Javolution - Java(TM) Solution for Real-Time and Embedded Systems
Copyright (c) 2006, Javolution (http://javolution.org)
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The libffi source distribution contains certain code that is not part
of libffi, and is only used as tooling to assist with the building and
testing of libffi. This includes the msvcc.sh script used to wrap the
Microsoft compiler with GNU compatible command-line options,
make_sunver.pl, and the libffi test code distributed in the
testsuite/libffi.bhaible directory. This code is distributed with
libffi for the purpose of convenience only, and libffi is in no way
derived from this code.

msvcc.sh an testsuite/libffi.bhaible are both distributed under the
terms of the GNU GPL version 2, as below.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This

General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains

a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but

does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to

control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then

the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals

of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.
libffi - Copyright (c) 1996-2020 Anthony Green, Red Hat, Inc and others.
See source files for details.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
``Software''), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ``AS IS'', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Apache Commons BeanUtils
Copyright 2000-2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
==========
==========
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed
their package names from 'org.apache.velocity' to 'clover.org.apache.velocity'. This was necessary to
avoid potential name conflicts during instrumentation of a code using the original library when using Clover.
No source code of the original library was modified.
==========
==========

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

/*
$Id: LICENSE.txt,v 1.1.1.1 2004/07/01 13:59:13 jvanzyl Exp $

Copyright 2002 (C) The Codehaus. All Rights Reserved.

Redistribution and use of this software and associated documentation
("Software"), with or without modification, are permitted provided
that the following conditions are met:

1. Redistributions of source code must retain copyright
statements and notices. Redistributions must also contain a

copy of this document.

2. Redistributions in binary form must reproduce the
above copyright notice, this list of conditions and the
following disclaimer in the documentation and/or other
materials provided with the distribution.

3. The name "classworlds" must not be used to endorse or promote
products derived from this Software without prior written
permission of The Codehaus. For written permission, please
contact bob@codehaus.org.

4. Products derived from this Software may not be called "classworlds"
nor may "classworlds" appear in their names without prior written
permission of The Codehaus. "classworlds" is a registered
trademark of The Codehaus.

5. Due credit should be given to The Codehaus.
(http://classworlds.codehaus.org/).

THIS SOFTWARE IS PROVIDED BY THE CODEHAUS AND CONTRIBUTORS
``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL
THE CODEHAUS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.

*/
==========
==========
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from the 'antlr.*' to the 'clover.antlr.*'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover.
No source code of the original library was modified.
==========
==========

SOFTWARE RIGHTS

ANTLR 1989-2006 Developed by Terence Parr

Partially supported by University of San Francisco & jGuru.com

We reserve no legal rights to the ANTLR--it is fully in the
public domain. An individual or company may do whatever
they wish with source code distributed with ANTLR or the
code generated by ANTLR, including the incorporation of
ANTLR, or its output, into commerical software.

We encourage users to develop software with ANTLR. However,
we do ask that credit is given to us for developing
ANTLR. By "credit", we mean that if you use ANTLR or
incorporate any source code into one of your programs
(commercial product, research project, or otherwise) that
you acknowledge this fact somewhere in the documentation,
research report, etc... If you like ANTLR and have
developed a nice tool with the output, please mention that
you developed it using ANTLR. In addition, we ask that the
headers remain intact in our source code. As long as these
guidelines are kept, we expect to continue enhancing this
system and expect to make other tools available as they are
completed.

The primary ANTLR guy:

Terence Parr
parrt@cs.usfca.edu
parrt@antlr.org
==========
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package name from 'com.google.common' to 'clover.com.google.common'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.
==========

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but

excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its

distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,

unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

==========
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package name from 'com.google.json' to 'clover.com.google.json'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.
==========

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work

(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses

granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"

replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

==========

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package name from 'com.lowagie' to 'clover.com.lowagie'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

==========

MOZILLA PUBLIC LICENSE
Version 1.1

---

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the
Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to
the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original
Code, prior Modifications used by a Contributor, and the Modifications
made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another

well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.
The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.
Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or

trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.
The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.
Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted

Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.
You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims.
If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.
If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.
Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.
You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.
You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.
You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

   If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

   This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

   6.1. New Versions.
   Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

   6.2. Effect of New Versions.
   Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

   6.3. Derivative Works.
   If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declatory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that

Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent

necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is _____.

The Initial Developer of the Original Code is _____.

Portions created by _____ are Copyright (C) _____
_____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[___] License"), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [___] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [___] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

Apache Velocity

Copyright (C) 2000-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
onoMnApeShTpQtDJbcUgJTIFONPQeUndIgfQWWNNddIwBl
mi2Kp5RjfhIJdGCSo<bOTNof2KNxm9KCi5lxEyKI9BJW3p
qOPQUXpopOopMMPqnPnXXQPNOPNRnqQNQqStwVxuQSTtVW
UrwSUSSSTVwxWSXNmrrpnmqmUUnpsvpntsmmmmmUUnpsvp
ntsmmmmmUUFmbkWJlroZbW4bsbilmjbkqUUnmmmm
Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name "DOM4J" must not be used to endorse or promote products derived from this Software without prior written

permission of MetaStuff, Ltd. For written permission,
please contact dom4j-info@metastuff.com.

4. Products derived from this Software may not be called "DOM4J"
nor may "DOM4J" appear in their names without prior written
permission of MetaStuff, Ltd. DOM4J is a registered
trademark of MetaStuff, Ltd.

5. Due credit should be given to the DOM4J Project -
http://www.dom4j.org

THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS
``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL
METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.
/*--

$Id: LICENSE.txt,v 1.11 2004/02/06 09:32:57 jhunter Exp $

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions, and the disclaimer that follows
these conditions in the documentation and/or other materials
provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products
derived from this software without prior written permission. For
written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor
may "JDOM" appear in their name, without prior written permission

from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the
end-user documentation provided with the redistribution and/or in the
software itself an acknowledgement equivalent to the following:
"This product includes software developed by the
JDOM Project (http://www.jdom.org/)."
Alternatively, the acknowledgment may be graphical using the logos
available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the JDOM Project and was originally
created by Jason Hunter <jhunter_AT_jdom_DOT_org> and
Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information
on the JDOM Project, please see <http://www.jdom.org/>.

*/
From: http://www.json.org/license.html
==========
==========

Copyright (c) 2002 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE

WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
SPDX-License-Identifier: Apache-2.0 OR LGPL-2.1

Java Native Access (JNA) is licensed under the LGPL, version 2.1
or later, or (from version 4.0 onward) the Apache License,
version 2.0.

You can freely decide which license you want to apply to the project.

You may obtain a copy of the LGPL License at:

http://www.gnu.org/licenses/licenses.html

A copy is also included in the downloadable source code package
containing JNA, in file "LGPL2.1", under the same directory
as this file.

You may obtain a copy of the Apache License at:

http://www.apache.org/licenses/

A copy is also included in the downloadable source code package
containing JNA, in file "AL2.0", under the same directory
as this file.

Commercial support may be available, please e-mail
twall[at]users[dot]sf[dot]net.
==========
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package name from 'org.apache.commons' to 'clover.org.apache.commons'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.
==========

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"

means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and

attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the

appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Copyright (c) 2005 - 2009 Taras Puchko
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.

/***
* CruiseControl, a Continuous Integration Toolkit
* Copyright (c) 2001-2003, ThoughtWorks, Inc.
* 651 W Washington Ave. Suite 500
* Chicago, IL 60661 USA
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* + Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* + Redistributions in binary form must reproduce the above
* copyright notice, this list of conditions and the following
* disclaimer in the documentation and/or other materials provided

```
*       with the distribution.
*
*     + Neither the name of ThoughtWorks, Inc., CruiseControl, nor the
*       names of its contributors may be used to endorse or promote
*       products derived from this software without specific prior
*       written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
* A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR
* CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
* PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
* LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
* NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
********************************************************************************/
```

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
http://www.atlassian.com/dms/wac/Atlassian_EULA_4-2.pdf
GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some
specially designated Free Software Foundation software, and to any
other libraries whose authors decide to use it. You can use it for
your libraries, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if
you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link a program with the library, you must provide
complete object files to the recipients so that they can relink them
with the library, after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright
the library, and (2) offer you this license which gives you legal
permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain
that everyone understands that there is no warranty for this free
library. If the library is modified by someone else and passed on, we
want its recipients to know that what they have is not the original
version, so that any problems introduced by others will not reflect on
the original authors' reputations.
.
Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that companies distributing free
software will individually obtain patent licenses, thus in effect
transforming the program into proprietary software. To prevent this,
we have made it clear that any patent must be licensed for everyone's
free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary
GNU General Public License, which was designed for utility programs. This
license, the GNU Library General Public License, applies to certain
designated libraries. This license is quite different from the ordinary
one; be sure to read it in full, and don't assume that anything in it is

the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a

table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

.

6. As an exception to the Sections above, you may also compile or

link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license

restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues),

conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by

the Free Software Foundation.

.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

/*--

$Id: LICENSE.txt,v 1.11 2004/02/06 09:32:57 jhunter Exp $

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions, and the disclaimer that follows
   these conditions in the documentation and/or other materials
   provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products
   derived from this software without prior written permission. For
   written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor
   may "JDOM" appear in their name, without prior written permission
   from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the
end-user documentation provided with the redistribution and/or in the
software itself an acknowledgement equivalent to the following:
    "This product includes software developed by the
     JDOM Project (http://www.jdom.org/)."
Alternatively, the acknowledgment may be graphical using the logos
available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the JDOM Project and was originally
created by Jason Hunter <jhunter_AT_jdom_DOT_org> and
Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information
on the JDOM Project, please see <http://www.jdom.org/>.

*/
Indiana University Extreme! Lab Software License

Version 1.1.1

Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in
  the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any,
  must include the following acknowledgment:

 "This product includes software developed by the Indiana University
 Extreme! Lab (http://www.extreme.indiana.edu/)."

Alternately, this acknowledgment may appear in the software itself,
if and wherever such third-party acknowledgments normally appear.

4. The names "Indiana Univeristy" and "Indiana Univeristy Extreme! Lab"
must not be used to endorse or promote products derived from this
software without prior written permission. For written permission,
please contact http://www.extreme.indiana.edu/.

5. Products derived from this software may not use "Indiana Univeristy"
name nor may "Indiana Univeristy" appear in their name, without prior
written permission of the Indiana University.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF
ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
==========
==========
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from the 'org.apache.commons' to the 'clover.org.apache.commons'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

==========

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

/*

File: Core.js

Description:

Provides common utility functions and the Class object used internally by the library.

Also provides the <TreeUtil> object for manipulating JSON tree structures

Some of the Basic utility functions and the Class system are based in the MooTools Framework <http://mootools.net>. Copyright (c) 2006-2009 Valerio Proietti, <http://mad4milk.net/>. MIT license <http://mootools.net/license.txt>.

Author:

Nicolas Garcia Belmonte

Copyright:

Copyright 2008-2009 by Nicolas Garcia Belmonte.

Homepage:

<http://thejit.org>

Version:

1.1.2

License:

BSD License

> Redistribution and use in source and binary forms, with or without
> modification, are permitted provided that the following conditions are met:
> * Redistributions of source code must retain the above copyright

> notice, this list of conditions and the following disclaimer.
> * Redistributions in binary form must reproduce the above copyright
> notice, this list of conditions and the following disclaimer in the
> documentation and/or other materials provided with the distribution.
> * Neither the name of the organization nor the
> names of its contributors may be used to endorse or promote products
> derived from this software without specific prior written permission.
>
> THIS SOFTWARE IS PROVIDED BY Nicolas Garcia Belmonte ``AS IS'' AND ANY
> EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
> WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
> DISCLAIMED. IN NO EVENT SHALL Nicolas Garcia Belmonte BE LIABLE FOR ANY
> DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
> (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
> LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
> ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
> (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
> SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.81 apiguardian-apiguardian-api 1.1.2

1.81.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.82 asm-commons 9.1

1.82.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bundle-DocURL: http://asm.ow2.org
Bundle-License: BSD-3-Clause;link=https://asm.ow2.io/LICENSE.txt
Bundle-ManifestVersion: 2
Bundle-Name: org.objectweb.asm.commons
Bundle-RequiredExecutionEnvironment: J2SE-1.5
Bundle-SymbolicName: org.objectweb.asm.commons
Bundle-Version: 9.1.0
Export-Package: org.objectweb.asm.commons;version="9.1";uses:="org.obj
ectweb.asm,org.objectweb.asm.signature,org.objectweb.asm.tree"
Implementation-Title: Usefull class adapters based on ASM, a very smal
l and fast Java bytecode manipulation framework
Implementation-Version: 9.1
Import-Package: org.objectweb.asm;version="[9.1,10)",org.objectweb.asm
.signature;version="[9.1,10)",org.objectweb.asm.tree;version="[9.1,10
)"
Module-Requires: org.objectweb.asm;transitive=true,org.objectweb.asm.t
ree;transitive=true,org.objectweb.asm.tree.analysis;transitive=true

Found in path(s):
* /opt/cola/permits/1183889887_1627494130.56/0/asm-commons-9-1-jar/META-INF/MANIFEST.MF

1.83 tomcat 9.0.65

1.83.1 Available under license :

Apache Commons Daemon
Copyright 1999-2022 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
Apache Tomcat Native Library
Copyright 2002-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code derived from netty-native
developed by the Netty project
(http://netty.io, https://github.com/netty/netty-tcnative/)
and from finagle-native developed at Twitter
(https://github.com/twitter/finagle).

This product includes software developed by the OpenSSL Project
for use in the OpenSSL Toolkit. (http://www.openssl.org/)

This product includes cryptographic software written by
Eric Young (eay@cryptsoft.com)

This product includes software written by
Tim Hudson (tjh@cryptsoft.com)
Apache Tomcat JDBC Pool
Copyright 2008-2022 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache Standard Taglib Specification API
Copyright 2001-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but

not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their

Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with

the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache Tomcat Native Library
Copyright 2002-2022 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code derived from netty-native
developed by the Netty project
(http://netty.io, https://github.com/netty/netty-tcnative/)
and from finagle-native developed at Twitter
(https://github.com/twitter/finagle).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but

excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its

distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,

unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

APACHE TOMCAT SUBCOMPONENTS:

Apache Tomcat includes a number of subcomponents with separate copyright notices and license terms. Your use of these subcomponents is subject to the terms and conditions of the following licenses.

For the following XML Schemas for Java EE Deployment Descriptors:
- javaee_5.xsd
- javaee_web_services_1_2.xsd
- javaee_web_services_client_1_2.xsd
- javaee_6.xsd
- javaee_web_services_1_3.xsd
- javaee_web_services_client_1_3.xsd
- jsp_2_2.xsd
- web-app_3_0.xsd
- web-common_3_0.xsd
- web-fragment_3_0.xsd
- javaee_7.xsd
- javaee_web_services_1_4.xsd
- javaee_web_services_client_1_4.xsd
- jsp_2_3.xsd
- web-app_3_1.xsd
- web-common_3_1.xsd
- web-fragment_3_1.xsd
- javaee_8.xsd
- web-app_4_0.xsd
- web-common_4_0.xsd
- web-fragment_4_0.xsd

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source

Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by

that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients. rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on

Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient.s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.
You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end

user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY.S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item,. as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys. fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation

which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For OpenSSL:

LICENSE ISSUES
==========

The OpenSSL toolkit stays under a double license, i.e. both the conditions of
the OpenSSL License and the original SSLeay license apply to the toolkit.
See below for the actual license texts.

OpenSSL License

/* ==

* Copyright (c) 1998-2018 The OpenSSL Project. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in
*   the documentation and/or other materials provided with the
*   distribution.
*
* 3. All advertising materials mentioning features or use of this
*   software must display the following acknowledgment:
*   "This product includes software developed by the OpenSSL Project
*   for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
*
* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
*   endorse or promote products derived from this software without
*   prior written permission. For written permission, please contact
*   openssl-core@openssl.org.
*
* 5. Products derived from this software may not be called "OpenSSL"
*   nor may "OpenSSL" appear in their names without prior written
*   permission of the OpenSSL Project.
*
* 6. Redistributions of any form whatsoever must retain the following
*   acknowledgment:
*   "This product includes software developed by the OpenSSL Project
*   for use in the OpenSSL Toolkit (http://www.openssl.org/)"
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.
* ==
*
* This product includes cryptographic software written by Eric Young

* (eay@cryptsoft.com). This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).
*
*/

Original SSLeay License

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
* All rights reserved.
*
* This package is an SSL implementation written
* by Eric Young (eay@cryptsoft.com).
* The implementation was written so as to conform with Netscapes SSL.
*
* This library is free for commercial and non-commercial use as long as
* the following conditions are aheared to. The following conditions
* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code. The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
*
* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
*   must display the following acknowledgement:
*   "This product includes cryptographic software written by
*   Eric Young (eay@cryptsoft.com)"
*   The word 'cryptographic' can be left out if the rouines from the library
*   being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
*   the apps directory (application code) you must include an acknowledgement:
*   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
*
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND

* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/
Apache Tomcat
Copyright 1999-2022 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

This software contains code derived from netty-native
developed by the Netty project
(https://netty.io, https://github.com/netty/netty-tcnative/)
and from finagle-native developed at Twitter
(https://github.com/twitter/finagle).

This software contains code derived from jgroups-kubernetes
developed by the JGroups project (http://www.jgroups.org/).

The Windows Installer is built with the Nullsoft
Scriptable Install System (NSIS), which is
open source software. The original software and
related information is available at
http://nsis.sourceforge.net.

Java compilation software for JSP pages is provided by the Eclipse
JDT Core Batch Compiler component, which is open source software.
The original software and related information is available at
https://www.eclipse.org/jdt/core/.

org.apache.tomcat.util.json.JSONParser.jj is a public domain javacc grammar
for JSON written by Robert Fischer.
https://github.com/RobertFischer/json-parser

For portions of the Tomcat JNI OpenSSL API and the OpenSSL JSSE integration
The org.apache.tomcat.jni and the org.apache.tomcat.net.openssl packages

are derivative work originating from the Netty project and the finagle-native
project developed at Twitter
* Copyright 2014 The Netty Project
* Copyright 2014 Twitter

For portions of the Tomcat cloud support
The org.apache.catalina.tribes.membership.cloud package contains derivative
work originating from the jgroups project.
https://github.com/jgroups-extras/jgroups-kubernetes
Copyright 2002-2018 Red Hat Inc.

The original XML Schemas for Java EE Deployment Descriptors:
- javaee_5.xsd
- javaee_web_services_1_2.xsd
- javaee_web_services_client_1_2.xsd
- javaee_6.xsd
- javaee_web_services_1_3.xsd
- javaee_web_services_client_1_3.xsd
- jsp_2_2.xsd
- web-app_3_0.xsd
- web-common_3_0.xsd
- web-fragment_3_0.xsd
- javaee_7.xsd
- javaee_web_services_1_4.xsd
- javaee_web_services_client_1_4.xsd
- jsp_2_3.xsd
- web-app_3_1.xsd
- web-common_3_1.xsd
- web-fragment_3_1.xsd
- javaee_8.xsd
- web-app_4_0.xsd
- web-common_4_0.xsd
- web-fragment_4_0.xsd

may be obtained from:
http://www.oracle.com/webfolder/technetwork/jsc/xml/ns/javaee/index.html
Apache Tomcat
Copyright 1999-2022 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The original XML Schemas for Java EE Deployment Descriptors:
- javaee_5.xsd
- javaee_web_services_1_2.xsd
- javaee_web_services_client_1_2.xsd
- javaee_6.xsd
- javaee_web_services_1_3.xsd

- javaee_web_services_client_1_3.xsd
- jsp_2_2.xsd
- web-app_3_0.xsd
- web-common_3_0.xsd
- web-fragment_3_0.xsd
- javaee_7.xsd
- javaee_web_services_1_4.xsd
- javaee_web_services_client_1_4.xsd
- jsp_2_3.xsd
- web-app_3_1.xsd
- web-common_3_1.xsd
- web-fragment_3_1.xsd
- javaee_8.xsd
- web-app_4_0.xsd
- web-common_4_0.xsd
- web-fragment_4_0.xsd

may be obtained from:
http://www.oracle.com/webfolder/technetwork/jsc/xml/ns/javaee/index.html

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

APACHE TOMCAT SUBCOMPONENTS:

Apache Tomcat includes a number of subcomponents with separate copyright notices
and license terms. Your use of these subcomponents is subject to the terms and
conditions of the following licenses.

For the Eclipse JDT Core Batch Compiler (ecj-x.x.x.jar) component:

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs

of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions)

under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

For the Windows Installer component:

* All NSIS source code, plug-ins, documentation, examples, header files and graphics, with the exception of the compression modules and where otherwise noted, are licensed under the zlib/libpng license.
* The zlib compression module for NSIS is licensed under the zlib/libpng license.
* The bzip2 compression module for NSIS is licensed under the bzip2 license.
* The lzma compression module for NSIS is licensed under the Common Public License version 1.0.

zlib/libpng license

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

bzip2 license

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,

this list of conditions and the following disclaimer.
2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, Cambridge, UK.

jseward@acm.org
Common Public License version 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not

derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal

actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such

provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Special exception for LZMA compression module

Igor Pavlov and Amir Szekely, the authors of the LZMA compression module for

NSIS, expressly permit you to statically or dynamically link your code (or bind by name) to the files from the LZMA compression module for NSIS without subjecting your linked code to the terms of the Common Public license version 1.0. Any modifications or additions to files from the LZMA compression module for NSIS, however, are subject to the terms of the Common Public License version 1.0.

For the following XML Schemas for Java EE Deployment Descriptors:
- javaee_5.xsd
- javaee_web_services_1_2.xsd
- javaee_web_services_client_1_2.xsd
- javaee_6.xsd
- javaee_web_services_1_3.xsd
- javaee_web_services_client_1_3.xsd
- jsp_2_2.xsd
- web-app_3_0.xsd
- web-common_3_0.xsd
- web-fragment_3_0.xsd
- javaee_7.xsd
- javaee_web_services_1_4.xsd
- javaee_web_services_client_1_4.xsd
- jsp_2_3.xsd
- web-app_3_1.xsd
- web-common_3_1.xsd
- web-fragment_3_1.xsd
- javaee_8.xsd
- web-app_4_0.xsd
- web-common_4_0.xsd
- web-fragment_4_0.xsd

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such

combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients. rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any

Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient.s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.
You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any

distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY.S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item,. as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys. fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against

the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

Apache Standard Taglib Implementation
Copyright 2001-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache Tomcat
Copyright 1999-2022 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.84 slf4j-api-module 1.7.36

1.84.1 Available under license :

No license file was found, but licenses were detected in source scan.

/**
* Copyright (c) 2004-2011 QOS.ch
* All rights reserved.
*
* Permission is hereby granted, free of charge, to any person obtaining
* a copy of this software and associated documentation files (the
* "Software"), to deal in the Software without restriction, including
* without limitation the rights to use, copy, modify, merge, publish,
* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to
* the following conditions:
*
* The above copyright notice and this permission notice shall be
* included in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
* LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
* OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*
*/

Found in path(s):
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/FormattingTuple.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/NOPLoggerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/spi/MDCAdapter.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/MarkerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/impl/StaticLoggerBinder.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/impl/StaticMarkerBinder.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/MDC.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/MessageFormatter.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/Marker.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-

jar/org/slf4j/helpers/BasicMDCAdapter.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/BasicMarker.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/ILoggerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/impl/StaticMDCBinder.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/MarkerIgnoringBase.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/NOPMDCAdapter.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/NamedLoggerBase.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/BasicMarkerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/Util.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/LoggerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/spi/LoggerFactoryBinder.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/spi/LocationAwareLogger.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/SubstituteLogger.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/Logger.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/spi/MarkerFactoryBinder.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/NOPLogger.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/IMarkerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/SubstituteLoggerFactory.java

1.85 zipkin-reporter-core 2.16.3

1.85.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.86 generex 1.0.2

1.86.1 Available under license :

No license file was found, but licenses were detected in source scan.

Generex is licensed under the Apache License, Version 2.0.
http://www.apache.org/licenses/LICENSE-2.0

Found in path(s):
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/README.md
No license file was found, but licenses were detected in source scan.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
 <modelVersion>4.0.0</modelVersion>
 <groupId>com.github.mifmif</groupId>
 <artifactId>generex</artifactId>
 <version>1.0.2</version>
 <name>Generex</name>
 <url>https://github.com/mifmif/Generex/tree/master</url>
 <description>Generex A Java Library for regex to Strings generation</description>
 <packaging>jar</packaging>
 <licenses>
  <license>
   <name>The Apache Software License, Version 2.0</name>
   <url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>
   <distribution>repo</distribution>
  </license>
 </licenses>
 <developers>
  <developer>
   <id>mifmif</id>
   <name>MIFRAH Youssef</name>
   <email>mifmif.com@gmail.com</email>
  </developer>
  <developer>
   <id>mkolisnyk</id>
   <name>mkolisnyk</name>
   <email>kolesnik.nickolay@gmail.com</email>
  </developer>
 </developers>

 <parent>
  <groupId>org.sonatype.oss</groupId>
  <artifactId>oss-parent</artifactId>
  <version>7</version>
```

```
</parent>

<properties>
 <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
 <maven.pmd.excludes>**/generated-sources/**</maven.pmd.excludes>
</properties>
<scm>
 <connection>scm:git:git@github.com:mifmif/Generex.git</connection>
 <developerConnection>scm:git:git@github.com:mifmif/Generex.git</developerConnection>
 <url>https://github.com/mifmif/Generex.git</url>
</scm>
<ciManagement>
 <url>https://travis-ci.org/mifmif/Generex/builds</url>
 <system>Travis</system>
 <notifiers>
  <notifier>
   <address>mifmif.com@gmail.com</address>
   <type>mail</type>
  </notifier>
  <notifier>
   <address>kolesnik.nickolay@gmail.com</address>
   <type>mail</type>
  </notifier>
 </notifiers>
</ciManagement>

<distributionManagement>
 <snapshotRepository>
  <id>ossrh</id>
  <url>https://oss.sonatype.org/content/repositories/snapshots</url>
 </snapshotRepository>
 <repository>
  <id>ossrh</id>
  <url>https://oss.sonatype.org/service/local/staging/deploy/maven2/</url>
 </repository>
</distributionManagement>


<build>
 <sourceDirectory>src/main/java</sourceDirectory>
 <testSourceDirectory>src/test/java</testSourceDirectory>
 <resources>
  <resource>
   <directory>src</directory>
   <excludes>
    <exclude>**/*.java</exclude>
   </excludes>
  </resource>
```

```
<resource>
 <directory>target/dependency</directory>
 <excludes>
  <exclude>**/*.java</exclude>
 </excludes>
</resource>
<resource>
 <directory>src/main/resources</directory>
 <includes>
  <include>**/*.properties</include>
 </includes>
</resource>
</resources>
<plugins>
 <plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-install-plugin</artifactId>
  <version>2.5.2</version>
  <configuration>
   <file>target/${project.artifactId}-${project.version}.jar</file>
   <groupId>${project.groupId}</groupId>
   <artifactId>${project.artifactId}</artifactId>
   <version>${project.version}</version>
   <packaging>maven-plugin</packaging>
  </configuration>
 </plugin>

 <plugin>
  <artifactId>maven-compiler-plugin</artifactId>
  <version>3.5.1</version>
  <configuration>
   <source>1.5</source>
   <target>1.5</target>
  </configuration>
 </plugin>
 <plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-jar-plugin</artifactId>
  <version>3.0.2</version>
  <configuration>
   <excludes>
    <exclude>*</exclude>
    <exclude>com/thoughtworks/**/*</exclude>
    <exclude>freemarker/**/*</exclude>
    <exclude>ftl/**/*</exclude>
    <exclude>i18n/**/*</exclude>
    <exclude>style/**/*</exclude>
    <exclude>junit/**/*</exclude>
```

```
   <exclude>licenses/**/*</exclude>
   <exclude>META-INF/maven/**/*</exclude>
   <exclude>org/codehaus/**/*</exclude>
   <exclude>org/hamcrest/**/*</exclude>
   <exclude>org/jbehave/**/*</exclude>
   <exclude>org/junit/**/*</exclude>
   <exclude>org/testng/**/*</exclude>
   <exclude>org/xmlpull/**/*</exclude>
   <exclude>stories/**/*</exclude>
   <exclude>style/**/*</exclude>
   <exclude>tests/**/*</exclude>
  </excludes>
  <archive>
   <manifest>
    <addClasspath>false</addClasspath>
    <addDefaultImplementationEntries>true</addDefaultImplementationEntries>
    <addDefaultSpecificationEntries>true</addDefaultSpecificationEntries>
    <addExtensions>false</addExtensions>
    <classpathLayoutType>simple</classpathLayoutType>
   </manifest>
  </archive>
 </configuration>
</plugin>
<plugin>
 <artifactId>maven-assembly-plugin</artifactId>
 <version>2.6</version>
 <configuration>
  <descriptors>
   <descriptor>./sources.xml</descriptor>
  </descriptors>
 </configuration>
</plugin>
<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-dependency-plugin</artifactId>
 <version>2.10</version>
</plugin>
<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-javadoc-plugin</artifactId>
 <version>2.10.3</version>
 <configuration>
  <show>private</show>
  <nohelp>false</nohelp>
  <failOnError>false</failOnError>
  <sourcepath>src/main/java</sourcepath>
 </configuration>
</plugin>
```

```
<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-release-plugin</artifactId>
 <configuration>
  <tagNameFormat>v@{project.version}</tagNameFormat>
  <preparationGoals>package install:install-file</preparationGoals>
  <scmCommentPrefix>#3</scmCommentPrefix>
 </configuration>
</plugin>
<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-deploy-plugin</artifactId>
 <version>2.8.2</version>
 <configuration>
  <skip>true</skip>
 </configuration>
</plugin>


<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-source-plugin</artifactId>
 <version>3.0.1</version>
 <executions>
  <execution>
   <goals>
    <goal>jar-no-fork</goal>
   </goals>
  </execution>
 </executions>
</plugin>


<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-javadoc-plugin</artifactId>
 <executions>
  <execution>
   <id>attach-javadocs</id>
   <goals>
    <goal>jar</goal>
   </goals>
  </execution>
 </executions>
</plugin>


<plugin>
```

```
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-gpg-plugin</artifactId>
 <executions>
  <execution>
   <id>sign-artifacts</id>
   <phase>deploy</phase>
   <goals>
    <goal>sign</goal>
   </goals>
  </execution>
 </executions>

</plugin>


<plugin>
 <groupId>org.sonatype.plugins</groupId>
 <artifactId>nexus-staging-maven-plugin</artifactId>
 <version>1.6.3</version>
 <executions>
  <execution>
   <id>default-deploy</id>
   <phase>deploy</phase>
   <goals>
    <goal>deploy</goal>
   </goals>
  </execution>
 </executions>
 <extensions>true</extensions>
 <configuration>
  <serverId>ossrh</serverId>
  <nexusUrl>https://oss.sonatype.org/</nexusUrl>
 </configuration>
</plugin>
<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-pmd-plugin</artifactId>
 <version>3.6</version>
 <configuration>
  <failOnViolation>true</failOnViolation>
  <excludeRoots>
   <excludeRoot>target/generated-sources/plugin</excludeRoot>
  </excludeRoots>
  <excludes>
   <exclude>**/HelpMojo.java</exclude>
  </excludes>
 </configuration>
 <executions>
```

```
 <execution>
  <goals>
   <goal>check</goal>
   <goal>cpd-check</goal>
  </goals>
  <configuration>
   <failOnViolation>true</failOnViolation>
   <excludes>
    <exclude>**/HelpMojo.java</exclude>
   </excludes>
  </configuration>
 </execution>
</executions>
</plugin>
<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-checkstyle-plugin</artifactId>
 <version>2.10</version>
 <configuration>
  <excludes>**/test/java/**/*.java</excludes>
  <configLocation>./sun_checks.xml</configLocation>
  <suppressionsLocation>./checkstyle-suppressions.xml</suppressionsLocation>
  <failOnViolation>true</failOnViolation>
 </configuration>
 <executions>
  <execution>
   <goals>
    <goal>check</goal>
   </goals>
  </execution>
 </executions>
</plugin>
<plugin>
 <groupId>org.codehaus.mojo</groupId>
 <artifactId>cobertura-maven-plugin</artifactId>
 <version>2.6</version>
 <configuration>
  <check>
   <haltOnFailure>false</haltOnFailure>
   <totalBranchRate>80</totalBranchRate>
   <totalLineRate>80</totalLineRate>
   <packageLineRate>80</packageLineRate>
   <packageBranchRate>80</packageBranchRate>
  </check>
  <instrumentation>
   <excludes>
    <exclude>**/HelpMojo.class</exclude>
   </excludes>
```

```
 </instrumentation>
</configuration>
<executions>
 <execution>
  <id>verification</id>
  <phase>verify</phase>
  <goals>
   <goal>check</goal>
   <goal>cobertura</goal>
  </goals>
 </execution>
</executions>
</plugin>
<plugin>
 <groupId>org.codehaus.mojo</groupId>
 <artifactId>javancss-maven-plugin</artifactId>
 <version>2.0</version>
 <configuration>
  <encoding>${project.build.sourceEncoding}</encoding>
  <ccnLimit>10</ccnLimit>
  <ncssLimit>100</ncssLimit>
  <failOnViolation>true</failOnViolation>
  <excludes>
   <exclude>**/tests/**/*.*</exclude>
  </excludes>
 </configuration>
 <executions>
  <execution>
   <goals>
    <goal>check</goal>
   </goals>
  </execution>
 </executions>
</plugin>
<plugin>
 <groupId>org.codehaus.mojo</groupId>
 <artifactId>findbugs-maven-plugin</artifactId>
 <version>3.0.3</version>
 <configuration>
  <encoding>${project.build.sourceEncoding}</encoding>
  <failOnError>true</failOnError>
  <excludeFilterFile>findBugsExclude.xml</excludeFilterFile>
 </configuration>
 <executions>
  <execution>
   <goals>
    <goal>check</goal>
   </goals>
```

```
    </execution>
   </executions>
  </plugin>
  <plugin>
   <groupId>org.codehaus.mojo</groupId>
   <artifactId>jdepend-maven-plugin</artifactId>
   <version>2.0</version>
  </plugin>
 </plugins>
</build>
<dependencies>
 <dependency>
  <groupId>dk.brics.automaton</groupId>
  <artifactId>automaton</artifactId>
  <version>1.11-8</version>
 </dependency>
 <dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>4.12</version>
  <scope>test</scope>
 </dependency>
 <dependency>
  <groupId>org.hamcrest</groupId>
  <artifactId>hamcrest-junit</artifactId>
  <version>2.0.0.0</version>
  <scope>test</scope>
 </dependency>
</dependencies>
</project>
```

Found in path(s):
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/pom.xml
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2014 y.mifrah
*

*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
```

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/util/Iterable.java
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/GenerexIterator.java
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/util/Iterator.java
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/Node.java
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/Main.java
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/Generex.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 y.mifrah
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/test/java/com/mifmif/common/regex/GenerexUnitTest.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 y.mifrah
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/test/java/com/mifmif/common/regex/GenerexIteratorUnitTest.java

1.87 jetbrains-annotations 13.0

1.87.1 Available under license :

Copyright 2000-2012 JetBrains s.r.o.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.88 fabric8-::-kubernetes-model-::-batch 5.12.1

1.88.1 Available under license :

No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340815699_1654860935.2749178/0/kubernetes-model-batch-5-12-1-sources-jar/manifest.vm
No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

       http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815699_1654860935.2749178/0/kubernetes-model-batch-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-batch/pom.xml

1.89 micrometer-metrics/micrometer 1.9.2

1.89.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/db/JooqExecuteListener.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/httpcomponents/HttpContextUtils.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/CumulativeHistogramLongTaskTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepTuple2.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/kafka/KafkaClientMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/okhttp3/OkHttpConnectionPoolMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/http/HttpServletRequestTagsProvider.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jetty/JettyClientMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/validate/PropertyValidator.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/httpcomponents/DefaultUriMapper.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jetty/JettyClientTags.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/MeterRegistryConfigValidator.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jetty/JettyClientTagsProvider.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/http/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/TimeWindowSum.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/kafka/KafkaMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/kafka/KafkaStreamsMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/db/MetricsDSLContext.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/http/Outcome.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-

jar/io/micrometer/core/instrument/binder/jpa/HibernateQueryMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/validate/InvalidReason.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/http/DefaultHttpServletRequestTagsProvider.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/cache/HazelcastIMapAdapter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/FixedBoundaryVictoriaMetricsHistogram.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/validate/Validated.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/validate/ValidationException.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/httpcomponents/MicrometerHttpClientInterceptor.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/commonspool2/CommonsObjectPool2Metrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/JvmHeapPressureMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jetty/JettySslHandshakeMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/http/HttpRequestTags.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/AbstractTimerBuilder.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/mongodb/MongoConnectionPoolTagsProvider.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/validate/DurationValidator.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepValue.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2018 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/push/PushMeterRegistry.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/ipc/http/HttpStatusClass.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/StrongReferenceGaugeFunction.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/MultiGauge.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/ipc/http/HttpUrlConnectionSender.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/kafka/KafkaConsumerMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/NamedThreadFactory.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/ipc/http/OkHttpSender.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/push/PushRegistryConfig.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/logging/LoggingRegistryConfig.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/logging/LoggingMeterRegistry.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/ipc/http/HttpSender.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012-2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/OnlyOnceLoggingDenyMeterFilter.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/package-info.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/**
* Copyright (c) 2004-2011 QOS.ch
* All rights reserved.
*
* Permission is hereby granted, free of charge, to any person obtaining
* a copy of this software and associated documentation files (the
* "Software"), to deal in the Software without restriction, including
* without limitation the rights to use, copy, modify, merge, publish,
* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to
* the following conditions:
*
* The above copyright notice and this permission notice shall be
* included in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
* LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
* OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/JdkLogger.java

* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/InternalLogger.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/logging/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/pause/ClockDriftPauseDetector.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeFunctionTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepMeasurement.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/FunctionTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/cache/JCacheMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeMeter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/annotation/Incubating.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/Metrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/hystrix/MicrometerMetricsPublisherThreadPool.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeTimeGauge.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/MeterEquivalence.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-

jar/io/micrometer/core/instrument/binder/logging/Log4j2Metrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/cumulative/CumulativeCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/DoubleFormat.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/cumulative/CumulativeTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/dropwizard/DropwizardFunctionCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/JvmCompilationMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/search/Search.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jersey/server/TimedFinder.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/search/RequiredSearch.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/noop/NoopFunctionTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/pause/PauseDetector.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepDouble.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/CountAtBucket.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/Statistic.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/DistributionSummary.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/dropwizard/DropwizardFunctionTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/BaseUnits.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/system/DiskSpaceMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/noop/NoopTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/simple/SimpleMeterRegistry.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeLongTaskTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-

jar/io/micrometer/core/instrument/dropwizard/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/noop/NoopGauge.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/noop/NoopDistributionSummary.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/ValueAtPercentile.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/StringEscapeUtils.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeCustomMeter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/AbstractMeter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jersey/server/MetricsRequestEventListener.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/Clock.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeMeterRegistry.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/dropwizard/DropwizardCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jpa/HibernateMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/Tag.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/aop/TimedAspect.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/Measurement.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/hystrix/MicrometerMetricsPublisher.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/hystrix/MicrometerMetricsPublisherCommand.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/dropwizard/DropwizardDistributionSummary.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepDistributionSummary.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/tomcat/TomcatMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/noop/NoopTimeGauge.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/Meter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/pause/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/TimedExecutorService.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-

jar/io/micrometer/core/instrument/binder/jetty/JettyStatisticsMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/search/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/AbstractPartition.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/MeterRegistry.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/cumulative/CumulativeDistributionSummary.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/TimeUtils.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/MeterPartition.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepMeterRegistry.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/cumulative/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/annotation/Timed.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/InvalidConfigurationException.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/MockClock.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/hystrix/HystrixMetricsBinder.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/DefaultGauge.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/HistogramSnapshot.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jersey/server/DefaultJerseyTagsProvider.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/noop/NoopCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jersey/server/JerseyTagsProvider.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/simple/CountingMode.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeDistributionSummary.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jersey/server/MetricsApplicationEventListener.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-

jar/io/micrometer/core/instrument/binder/jersey/server/AnnotationFinder.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/Histogram.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/cache/CaffeineCacheMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/TimedExecutor.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/AbstractCompositeMeter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/db/PostgreSQLDatabaseMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/pause/NoPauseDetector.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/IOUtils.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/noop/NoopFunctionCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/NamingConvention.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/ClassLoaderMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/system/UptimeMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/ImmutableTag.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/TimeWindowFixedBoundaryHistogram.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepFunctionTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/cache/CacheMeterBinder.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/okhttp3/OkHttpMetricsEventListener.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/AbstractTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/AbstractDistributionSummary.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/cumulative/CumulativeFunctionCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/HistogramGauges.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepFunctionCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/MeterFilter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-

jar/io/micrometer/core/instrument/TimeGauge.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/Tags.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/noop/NoopMeter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/dropwizard/DropwizardRate.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/JvmThreadMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/noop/NoopLongTaskTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/DiskSpaceMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/dropwizard/DropwizardClock.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/aop/CountedAspect.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jersey/server/JerseyTags.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/PercentileHistogramBuckets.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepRegistryConfig.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/cache/HazelcastCacheMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/push/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/system/ProcessorMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/dropwizard/DropwizardConfig.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/HierarchicalNameMapper.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/DefaultLongTaskTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/FunctionCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/util/StringUtils.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/AbstractTimeWindowHistogram.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/DefaultMeter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/Gauge.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-

jar/io/micrometer/core/instrument/config/MeterFilterReply.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/annotation/Counted.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/DistributionStatisticConfig.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/db/DatabaseTableMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/MissingRequiredConfigurationException.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/annotation/TimedSet.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/MeterBinder.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/system/FileDescriptorMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/simple/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeFunctionCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/TimeWindowMax.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/cache/GuavaCacheMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/HistogramSupport.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/step/StepLong.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/cache/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/Mergeable.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/cumulative/CumulativeFunctionTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/JvmGcMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/TimeWindowPercentileHistogram.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/distribution/NoopHistogram.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/search/MeterNotFoundException.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/dropwizard/DropwizardTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/dropwizard/DropwizardGauge.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-

jar/io/micrometer/core/instrument/binder/logging/LogbackMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/LongTaskTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/ExecutorServiceMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeGauge.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/simple/SimpleConfig.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/cache/EhCache2Metrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/MeterRegistryConfig.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/noop/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/composite/CompositeTimer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/dropwizard/DropwizardMeterRegistry.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/Counter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/JvmMemoryMetrics.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/InternalLogLevel.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/AbstractInternalLogger.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/JdkLoggerFactory.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/InternalLoggerFactory.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/Slf4JLogger.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/Slf4JLoggerFactory.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jetty/JettyConnectionMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/TimedCallable.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/httpcomponents/PoolingHttpClientConnectionManagerMetricsBinder.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/mongodb/MongoMetricsConnectionPoolListener.java

* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/TimedScheduledExecutorService.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/mongodb/MongoMetricsCommandListener.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jetty/OnCompletionAsyncListener.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jetty/JettyServerThreadPoolMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/JvmMemory.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/WarnThenDebugLogger.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/internal/TimedRunnable.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jetty/InstrumentedQueuedThreadPool.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/Timer.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/httpcomponents/MicrometerHttpRequestExecutor.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jetty/TimedHandler.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2002-2017 the original author or authors.
* Copyright 2017-2021 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/lang/NonNullApi.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/lang/Nullable.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/lang/NonNullFields.java

* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/lang/NonNull.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2019 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

```
/*
* Copyright 2017 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/LocationAwareSlf4JLogger.java
No license file was found, but licenses were detected in source scan.

```
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/META-INF/native-image/io.micrometer/micrometer-core/native-image.properties

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2021 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/grpc/MetricCollectingClientInterceptor.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/grpc/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/mongodb/DefaultMongoCommandTagsProvider.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/cache/CaffeineStatsCounter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/grpc/MetricCollectingServerInterceptor.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/grpc/MetricCollectingClientCall.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/grpc/MetricCollectingServerCallListener.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/grpc/MetricCollectingClientCallListener.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/JvmInfoMetrics.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/mongodb/DefaultMongoConnectionPoolTagsProvider.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/mongodb/MongoCommandTagsProvider.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/grpc/AbstractMetricCollectingInterceptor.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/grpc/MetricCollectingServerCall.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2022 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jpa/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/db/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jvm/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jersey/server/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/lang/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/hystrix/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/aop/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/ipc/http/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/logging/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/config/validate/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/kafka/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/tomcat/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/httpcomponents/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/jetty/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/commonspool2/package-info.java

* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/mongodb/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/system/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/instrument/binder/okhttp3/package-info.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/annotation/package-info.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2019 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/**
* Copyright (c) 2004-2011 QOS.ch
* All rights reserved.
*
* Permission is hereby granted, free  of charge, to any person obtaining
* a  copy  of this  software  and  associated  documentation files  (the
* "Software"), to  deal in  the Software without  restriction, including
* without limitation  the rights to  use, copy, modify,  merge, publish,
```

* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to
* the following conditions:
*
* The above copyright notice and this permission notice shall be
* included in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
* LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
* OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*
*/

Found in path(s):
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/MessageFormatter.java
* /opt/cola/permits/1519137050_1671720309.9982727/0/micrometer-core-1-9-2-sources-jar/io/micrometer/core/util/internal/logging/FormattingTuple.java

1.90 spring-boot-test 2.7.2

1.90.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Spring Boot 2.7.2
Copyright (c) 2012-2022 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

1.91 hdrhistogram 2.1.12

1.91.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bnd-LastModified: 1575980548657
Build-Jdk: 1.8.0_232
Built-By: gil
Bundle-Description: HdrHistogram supports the recording and analyzing sa
mpled data value counts across a configurable integer value rang
e with configurable value precision within the range. Value prec
ision is expressed as the number of significant digits in the va
lue recording, and provides control over value quantization beha
vior across the value range and the subsequent value resolution
at any given level.
Bundle-License: http://creativecommons.org/publicdomain/zero/1.0/, https
://opensource.org/licenses/BSD-2-Clause
Bundle-ManifestVersion: 2
Bundle-Name: HdrHistogram
Bundle-SymbolicName: org.hdrhistogram.HdrHistogram
Bundle-Version: 2.1.12
Created-By: Apache Maven Bundle Plugin
Export-Package: org.HdrHistogram;version="2.1.12",org.HdrHistogram.packe
darray;version="2.1.12"
Implementation-Title: HdrHistogram
Implementation-Vendor-Id: org.hdrhistogram
Implementation-Version: 2.1.12
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.7))"
Specification-Title: HdrHistogram
Specification-Version: 2.1.12
Tool: Bnd-2.3.0.201405100607

Found in path(s):
* /opt/cola/permits/1185944801_1628011594.04/0/hdrhistogram-2-1-12-3-jar/META-INF/MANIFEST.MF
No license file was found, but licenses were detected in source scan.

* public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/
<name>Public Domain, per Creative Commons CC0</name>

Found in path(s):
* /opt/cola/permits/1185944801_1628011594.04/0/hdrhistogram-2-1-12-3-jar/META-
INF/maven/org.hdrhistogram/HdrHistogram/pom.xml

1.92 kotlin 1.6.21

1.92.1 Available under license :

No license file was found, but licenses were detected in source scan.

{"version":3,"file":"kotlin.js","sources":["wrapper.js","js/arrayUtils.js","js/callableReferenceUtils.js","js/conversions.js","js/core.js","js/long.js","js/markerFunctions.js","js/misc.js","js/polyfills.js","js/rtti.js","runtime/arrayUtils.kt","runtime/Enum.kt","primitiveCompanionObjects.kt","common/src/generated/_Arrays.kt","common/src/generated/_Ranges.kt","unsigned/src/kotlin/UByte.kt","unsigned/src/kotlin/UInt.kt","unsigned/src/kotlin/UShort.kt","builtin-sources/Ranges.kt","src/kotlin/collections/Collections.kt","src/kotlin/collections/Maps.kt","src/kotlin/collections/Sets.kt","src/kotlin/text/StringNumberConversions.kt","src/kotlin/time/Duration.kt","unsigned/src/kotlin/UnsignedUtils.kt","src/kotlin/collections/Iterables.kt","src/kotlin/collections/Sequences.kt","src/kotlin/util/Preconditions.kt","js/src/generated/_ArraysJs.kt","src/kotlin/comparisons/Comparisons.kt","src/kotlin/util/Standard.kt","js/src/generated/_ComparisonsJs.kt","unsigned/src/kotlin/ULong.kt","common/src/generated/_Collections.kt","js/src/kotlin/collections.kt","src/kotlin/collections/Iterators.kt","common/src/generated/_Comparisons.kt","common/src/generated/_Maps.kt","common/src/generated/_OneToManyTitlecaseMappings.kt","js/src/kotlin/text/char.kt","js/src/kotlin/text/string.kt","src/kotlin/text/Char.kt","src/kotlin/CharCode.kt","common/src/generated/_Sequences.kt","common/src/generated/_Sets.kt","common/src/generated/_Strings.kt","src/kotlin/text/Strings.kt","unsigned/src/kotlin/UByteArray.kt","unsigned/src/kotlin/UIntArray.kt","unsigned/src/kotlin/ULongArray.kt","unsigned/src/kotlin/UShortArray.kt","common/src/generated/_UArrays.kt","common/src/generated/_UCollections.kt","common/src/generated/_UComparisons.kt","common/src/generated/_URanges.kt","common/src/generated/_USequences.kt","common/src/kotlin/ExceptionsH.kt","common/src/kotlin/JsAnnotationsH.kt","common/src/kotlin/ioH.kt","builtin-sources/Collections.kt","builtin-sources/Iterators.kt","builtin-sources/ProgressionIterators.kt","builtin-sources/Progressions.kt","builtin-sources/Range.kt","builtin-sources/Unit.kt","builtin-sources/annotation/Annotations.kt","builtin-sources/internal/InternalAnnotations.kt","builtin-sources/internal/progressionUtil.kt","src/kotlin/builtins.kt","src/kotlin/jsTypeOf.kt","src/kotlin/kotlin.kt","src/kotlin/charCode_js-v1.kt","src/kotlin/coroutines/CoroutineImpl.kt","src/kotlin/util/Result.kt","src/kotlin/coroutines/Continuation.kt","src/kotlin/coroutines/intrinsics/IntrinsicsJs.kt","src/kotlin/currentBeMisc.kt","src/kotlin/exceptions.kt","src/kotlin/jsOperators.kt","src/kotlin/math_js-v1.kt","src/kotlin/numbers_js-v1.kt","src/kotlin/reflection_js-v1.kt","src/kotlin/text/numberConversions_js-v1.kt","js/src/generated/_CharCategories.kt","js/src/generated/_CollectionsJs.kt","js/src/generated/_DigitChars.kt","js/src/generated/_LetterChars.kt","js/src/generated/_OtherLowercaseChars.kt","js/src/generated/_OtherUppercaseChars.kt","js/src/generated/_StringsJs.kt","js/src/generated/_TitlecaseMappings.kt","js/src/generated/_UArraysJs.kt","js/src/generated/_WhitespaceChars.kt","js/src/kotlin/Comparator.kt","js/src/kotlin/annotations.kt","js/src/kotlin/annotationsJVM.kt","js/src/kotlin/collections/AbstractMutableCollection.kt","js/src/kotlin/collections/AbstractMutableList.kt","js/src/kotlin/collections/AbstractMutableMap.kt","js/src/kotlin/collections/AbstractMutableSet.kt","js/src/kotlin/collections/ArrayList.kt","js/src/kotlin/collections/ArraySorting.kt","js/src/kotlin/collections/ArraysJs.kt","js/src/kotlin/collections/EqualityComparator.kt","js/src/kotlin/collections/HashMap.kt","js/src/kotlin/collections/HashSet.kt","js/src/kotlin/collections/InternalHashCodeMap.kt","js/src/kotlin/collections/InternalMap.kt","js/src/kotlin/collections/InternalStringMap.kt","js/src/kotlin/collections/LinkedHashMap.kt","js/src/kotlin/collections/LinkedHashSet.kt","js/src/kotlin/concurrent.kt","js/src/kotlin/console.kt","js/src/kotlin/coroutines/SafeContinuationJs.kt","js/src/kotlin/coroutines/cancellation/CancellationException.kt","js/src/kotlin/coroutines/js/internal/EmptyContinuation.kt","js/src/kotlin/date.kt","js/src/kotlin/dom/Builders.kt","js/src/kotlin/dom/Classes.kt","js/src/kotlin/dom/Dom.kt","js/src/kotlin/dom/EventListener.kt","js/src/kotlin/dom/ItemArrayLike.kt","js/src/kotlin/dom/Mutations.kt","js/src/kotlin/dynamic.kt","js/src/kotlin/exceptionUtils.kt","js/src/kotlin/grouping.kt","src/kotlin/collections/Grouping.kt","js/src/kotlin/internalAnnotations.kt","js/src/kotlin/json.kt","js/src/kotlin/math.kt","js/src/kotlin/numbers.kt","js/src/kotlin/promise.kt","js/src/kotlin/random/PlatformRandom.kt","js/src/kotlin/reflect/AssociatedObjects.kt","js/src/kotlin/reflect/JsClass.kt","js/src/kotlin/reflect/KClassImpl.kt","js/src/kotlin/reflect/KClassesImpl.kt","js/src/kotlin/reflect/KTypeHelpers.kt","js/src/kotlin/reflect/KTypeImpl.kt","js/src/kotlin/reflect/KTypeParameterImpl.kt","js/src/kotlin/reflect/primitiv

es.kt","js/src/kotlin/reflect/reflection.kt","js/src/kotlin/regexp.kt","js/src/kotlin/sequence.kt","js/src/kotlin/text/CharCategoryJS.kt","js/src/kotlin/text/CharacterCodingExceptionJs.kt","js/src/kotlin/text/StringBuilderJs.kt","js/src/kotlin/text/numberConversions.kt","js/src/kotlin/text/regex.kt","src/kotlin/text/StringBuilder.kt","js/src/kotlin/text/stringsCode.kt","js/src/kotlin/text/utf8Encoding.kt","js/src/kotlin/throwableExtensions.kt","js/src/kotlin/time/DurationJs.kt","js/src/kotlin/time/DurationUnit.kt","js/src/kotlin/time/MonoTimeSource.kt","js/src/kotlinx/dom/Builders.kt","js/src/kotlinx/dom/Classes.kt","src/kotlin/text/regex/RegexExtensions.kt","js/src/kotlinx/dom/Dom.kt","js/src/kotlinx/dom/Mutations.kt","js/src/org.w3c/deprecated.kt","js/src/org.w3c/org.khronos.webgl.kt","js/src/org.w3c/org.w3c.dom.clipboard.kt","js/src/org.w3c/org.w3c.dom.css.kt","js/src/org.w3c/org.w3c.dom.encryptedmedia.kt","js/src/org.w3c/org.w3c.dom.events.kt","js/src/org.w3c/org.w3c.dom.kt","js/src/org.w3c/org.w3c.fetch.kt","js/src/org.w3c/org.w3c.dom.mediacapture.kt","js/src/org.w3c/org.w3c.dom.mediasource.kt","js/src/org.w3c/org.w3c.dom.pointerevents.kt","js/src/org.w3c/org.w3c.dom.svg.kt","js/src/org.w3c/org.w3c.files.kt","js/src/org.w3c/org.w3c.notifications.kt","js/src/org.w3c/org.w3c.workers.kt","js/src/org.w3c/org.w3c.xhr.kt","src/kotlin/annotations/Experimental.kt","src/kotlin/annotations/ExperimentalStdlibApi.kt","src/kotlin/annotations/Inference.kt","src/kotlin/annotations/Multiplatform.kt","src/kotlin/annotations/OptIn.kt","src/kotlin/collections/AbstractCollection.kt","src/kotlin/collections/AbstractIterator.kt","src/kotlin/collections/AbstractList.kt","src/kotlin/collections/AbstractMap.kt","src/kotlin/collections/AbstractSet.kt","src/kotlin/collections/ArrayDeque.kt","src/kotlin/collections/Arrays.kt","src/kotlin/collections/BrittleContainsOptimization.kt","src/kotlin/collections/IndexedValue.kt","src/kotlin/collections/MapAccessors.kt","src/kotlin/collections/MapWithDefault.kt","src/kotlin/collections/MutableCollections.kt","src/kotlin/collections/ReversedViews.kt","src/kotlin/collections/SequenceBuilder.kt","src/kotlin/collections/SlidingWindow.kt","src/kotlin/collections/UArraySorting.kt","src/kotlin/comparisons/compareTo.kt","src/kotlin/contracts/ContractBuilder.kt","src/kotlin/coroutines/ContinuationInterceptor.kt","src/kotlin/coroutines/CoroutineContext.kt","src/kotlin/coroutines/CoroutineContextImpl.kt","src/kotlin/coroutines/intrinsics/Intrinsics.kt","src/kotlin/experimental/bitwiseOperations.kt","src/kotlin/experimental/inferenceMarker.kt","src/kotlin/internal/Annotations.kt","src/kotlin/properties/Delegates.kt","src/kotlin/properties/Interfaces.kt","src/kotlin/properties/ObservableProperty.kt","src/kotlin/properties/PropertyReferenceDelegates.kt","src/kotlin/random/Random.kt","src/kotlin/random/URandom.kt","src/kotlin/random/XorWowRandom.kt","src/kotlin/ranges/Ranges.kt","src/kotlin/reflect/KClasses.kt","src/kotlin/reflect/KTypeProjection.kt","src/kotlin/reflect/KVariance.kt","src/kotlin/reflect/typeOf.kt","src/kotlin/text/Appendable.kt","src/kotlin/text/Indent.kt","src/kotlin/text/Typography.kt","src/kotlin/text/regex/MatchResult.kt","src/kotlin/time/DurationUnit.kt","src/kotlin/time/ExperimentalTime.kt","src/kotlin/time/TimeSource.kt","src/kotlin/time/TimeSources.kt","src/kotlin/time/measureTime.kt","src/kotlin/util/DeepRecursive.kt","src/kotlin/util/FloorDivMod.kt","src/kotlin/util/HashCode.kt","src/kotlin/util/KotlinVersion.kt","src/kotlin/util/Lateinit.kt","src/kotlin/util/Lazy.kt","src/kotlin/util/Numbers.kt","src/kotlin/util/Suspend.kt","src/kotlin/util/Tuples.kt","unsigned/src/kotlin/UIntRange.kt","unsigned/src/kotlin/UIterators.kt","unsigned/src/kotlin/ULongRange.kt","unsigned/src/kotlin/UMath.kt","unsigned/src/kotlin/UNumbers.kt","unsigned/src/kotlin/UProgressionUtil.kt","unsigned/src/kotlin/UStrings.kt","unsigned/src/kotlin/annotations/Unsigned.kt","common/src/kotlin/MathH.kt"],"sourcesContent":["(function (root, factory) {\n if (typeof define === 'function' && define.amd) {\n define('kotlin', ['exports'], factory);\n }\n else if (typeof exports === 'object') {\n factory(module.exports);\n }\n else {\n root.kotlin = {};\n factory(root.kotlin);\n }\n}(this, function (Kotlin) {\n var _ = Kotlin;\n\n insertContent();\n}));\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\nKotlin.isBooleanArray = function (a) {\n return (Array.isArray(a) || a instanceof Int8Array) && a.$type$ === \"BooleanArray\"\n};\n\nKotlin.isByteArray = function (a) {\n return a instanceof Int8Array && a.$type$!== \"BooleanArray\"\n};\n\nKotlin.isShortArray = function (a) {\n return a instanceof Int16Array\n};\n\nKotlin.isCharArray = function (a) {\n return a instanceof Uint16Array && a.$type$ === \"CharArray\"\n};\n\nKotlin.isIntArray = function (a) {\n return a instanceof Int32Array\n};\n\nKotlin.isFloatArray = function (a) {\n return a instanceof Float32Array\n};\n\nKotlin.isDoubleArray = function (a) {\n return a instanceof Float64Array\n};\n\nKotlin.isLongArray = function (a) {\n return Array.isArray(a) && a.$type$ === \"LongArray\"\n};\n\nKotlin.isArray = function (a) {\n return Array.isArray(a) &&

```
!a.$type$;\n};\n\nKotlin.isArrayish = function (a) {\n    return Array.isArray(a) ||
ArrayBuffer.isView(a)\n};\n\nKotlin.arrayToString = function (a) {\n    if (a === null) return \"null\"\n    var
toString = Kotlin.isCharArray(a) ? String.fromCharCode : Kotlin.toString;\n    return \"[\" +
Array.prototype.map.call(a, function(e) { return toString(e); }).join(\", \") + \"]\";\n};\n\nKotlin.arrayDeepToString
= function (arr) {\n    return Kotlin.kotlin.collections.contentDeepToStringImpl(arr);\n};\n\nKotlin.arrayEquals =
function (a, b) {\n    if (a === b) {\n        return true;\n    }\n    if (a === null || b === null || !Kotlin.isArrayish(b) ||
a.length !== b.length) {\n        return false;\n    }\n\n    for (var i = 0, n = a.length; i < n; i++) {\n        if
(!Kotlin.equals(a[i], b[i])) {\n            return false;\n        }\n    }\n    return true;\n};\n\nKotlin.arrayDeepEquals =
function (a, b) {\n    return Kotlin.kotlin.collections.contentDeepEqualsImpl(a, b);\n};\n\nKotlin.arrayHashCode =
function (arr) {\n    if (arr === null) return 0\n    var result = 1;\n    for (var i = 0, n = arr.length; i < n; i++) {\n
result = ((31 * result | 0) + Kotlin.hashCode(arr[i])) | 0;\n    }\n    return result;\n};\n\nKotlin.arrayDeepHashCode =
function (arr) {\n    return
Kotlin.kotlin.collections.contentDeepHashCodeImpl(arr);\n};\n\nKotlin.primitiveArraySort = function (array) {\n
array.sort(Kotlin.doubleCompareTo)\n};\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors. \n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\n */\n\nKotlin.getCallableRef = function(name, f) {\n    f.callableName = name;\n    return
f;\n};\n\nKotlin.getPropertyCallableRef = function(name, paramCount, getter, setter) {\n    getter.get = getter;\n
getter.set = setter;\n    getter.callableName = name;\n    return getPropertyRefClass(getter, setter,
propertyRefClassMetadataCache[paramCount]);\n};\n\nfunction getPropertyRefClass(obj, setter, cache) {\n
obj.$metadata$ = getPropertyRefMetadata(typeof setter === \"function\" ? cache.mutable : cache.immutable);\n
obj.constructor = obj;\n    return obj;\n}\n\nvar propertyRefClassMetadataCache = [\n    {\n        mutable: { value:
null, implementedInterface: function () {\n            return Kotlin.kotlin.reflect.KMutableProperty0 }\n        },\n
immutable: { value: null, implementedInterface: function () {\n            return Kotlin.kotlin.reflect.KProperty0 }\n
    }\n    },\n    {\n        mutable: { value: null, implementedInterface: function () {\n            return
Kotlin.kotlin.reflect.KMutableProperty1 }\n        },\n        immutable: { value: null, implementedInterface: function
() {\n            return Kotlin.kotlin.reflect.KProperty1 }\n        }\n    }\n];\n\nfunction getPropertyRefMetadata(cache)
{\n    if (cache.value === null) {\n        cache.value = {\n            interfaces: [cache.implementedInterface()],\n
baseClass: null,\n            functions: {},\n            properties: {},\n            types: {},\n            staticMembers: {}\n
};\n    }\n    return cache.value;\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors. \n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\n */\n\nKotlin.toShort = function (a) {\n    return (a & 0xFFFF) << 16 >>
16;\n};\n\nKotlin.toByte = function (a) {\n    return (a & 0xFF) << 24 >> 24;\n};\n\nKotlin.toChar = function (a) {\n
  return a & 0xFFFF;\n};\n\nKotlin.numberToLong = function (a) {\n    return a instanceof Kotlin.Long ? a :
Kotlin.Long.fromNumber(a);\n};\n\nKotlin.numberToInt = function (a) {\n    return a instanceof Kotlin.Long ?
a.toInt() : Kotlin.doubleToInt(a);\n};\n\nKotlin.numberToShort = function (a) {\n    return
Kotlin.toShort(Kotlin.numberToInt(a));\n};\n\nKotlin.numberToByte = function (a) {\n    return
Kotlin.toByte(Kotlin.numberToInt(a));\n};\n\nKotlin.numberToDouble = function (a) {\n    return
+a;\n};\n\nKotlin.numberToChar = function (a) {\n    return
Kotlin.toChar(Kotlin.numberToInt(a));\n};\n\nKotlin.doubleToInt = function(a) {\n    if (a > 2147483647) return
2147483647;\n    if (a < -2147483648) return -2147483648;\n    return a | 0;\n};\n\nKotlin.toBoxedChar = function
(a) {\n    if (a == null) return a;\n    if (a instanceof Kotlin.BoxedChar) return a;\n    return new
Kotlin.BoxedChar(a);\n};\n\nKotlin.unboxChar = function(a) {\n    if (a == null) return a;\n    return
Kotlin.toChar(a);\n};\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors. \n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\n */\n\nKotlin.equals = function (obj1, obj2) {\n    if (obj1 == null) {\n        return obj2 ==
null;\n    }\n\n    if (obj2 == null) {\n        return false;\n    }\n\n    if (obj1 !== obj1) {\n        return obj2 !== obj2;\n
}\n\n    if (typeof obj1 === \"object\" && typeof obj1.equals === \"function\") {\n        return obj1.equals(obj2);\n
}\n\n    if (typeof obj1 === \"number\" && typeof obj2 === \"number\") {\n        return obj1 === obj2 && (obj1 !==
```

0 || 1 / obj1 === 1 / obj2)\n }\n\n return obj1 === obj2;\n};\n\nKotlin.hashCode = function (obj) {\n if (obj == null) {\n   return 0;\n }\n var objType = typeof obj;\n if (\"object\" === objType) {\n   return \"function\" === typeof obj.hashCode ? obj.hashCode() : getObjectHashCode(obj);\n }\n if (\"function\" === objType) {\n   return getObjectHashCode(obj);\n }\n if (\"number\" === objType) {\n   return Kotlin.numberHashCode(obj);\n }\n if (\"boolean\" === objType) {\n   return Number(obj)\n }\n\n var str = String(obj);\n return getStringHashCode(str);\n};\n\n\nKotlin.toString = function (o) {\n if (o == null) {\n   return \"null\";\n }\n else if (Kotlin.isArrayish(o)) {\n   return \"[...]\";\n }\n else {\n   return o.toString();\n }\n};\n\n/** @const */\nvar POW_2_32 = 4294967296;\n// TODO: consider switching to Symbol type once we are on ES6.\n/** @const */\nvar OBJECT_HASH_CODE_PROPERTY_NAME = \"kotlinHashCodeValue$\";\n\nfunction getObjectHashCode(obj) {\n if (!(OBJECT_HASH_CODE_PROPERTY_NAME in obj)) {\n   var hash = (Math.random() * POW_2_32) | 0; // Make 32-bit singed integer.\n   Object.defineProperty(obj, OBJECT_HASH_CODE_PROPERTY_NAME, { value: hash, enumerable: false });\n }\n return obj[OBJECT_HASH_CODE_PROPERTY_NAME];\n}\n\nfunction getStringHashCode(str) {\n var hash = 0;\n for (var i = 0; i < str.length; i++) {\n   var code = str.charCodeAt(i);\n   hash = (hash * 31 + code) | 0; // Keep it 32-bit.\n }\n return hash;\n}\n\nKotlin.identityHashCode = getObjectHashCode;\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Copyright 2009 The Closure Library Authors. All Rights Reserved.\n//\n// Licensed under the Apache License, Version 2.0 (the \"License\");\n// you may not use this file except in compliance with the License.\n// You may obtain a copy of the License at\n//\n// http://www.apache.org/licenses/LICENSE-2.0\n//\n// Unless required by applicable law or agreed to in writing, software\n// distributed under the License is distributed on an \"AS-IS\" BASIS,\n// WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.\n\n/**\n * Constructs a 64-bit two's-complement integer, given its low and high 32-bit\n * values as *signed* integers. See the from* functions below for more\n * convenient ways of constructing Longs.\n *\n * The internal representation of a long is the two given signed, 32-bit values.\n * We use 32-bit pieces because these are the size of integers on which\n * Javascript performs bit-operations. For operations like addition and\n * multiplication, we split each number into 16-bit pieces, which can easily be\n * multiplied within Javascript's floating-point representation without overflow\n * or change in sign.\n *\n * In the algorithms below, we frequently reduce the negative case to the\n * positive case by negating the input(s) and then post-processing the result.\n * Note that we must ALWAYS check specially whether those values are MIN_VALUE\n * (-2^63) because -MIN_VALUE == MIN_VALUE (since 2^63 cannot be represented as\n * a positive number, it overflows back into a negative). Not handling this\n * case would often result in infinite recursion.\n *\n * @param {number} low The low (signed) 32 bits of the long.\n * @param {number} high The high (signed) 32 bits of the long.\n * @constructor\n * @final\n */\nKotlin.Long = function(low, high) {\n /**\n   * @type {number}\n   * @private\n   */\n this.low_ = low | 0; // force into 32 signed bits.\n\n /**\n   * @type {number}\n   * @private\n   */\n this.high_ = high | 0; // force into 32 signed bits.\n};\n\nKotlin.Long.$metadata$ = {\n kind: \"class\",\n simpleName: \"Long\",\n interfaces:[]\n};\n\n\n// NOTE: Common constant values ZERO, ONE, NEG_ONE, etc. are defined below the\n// from* methods on which they depend.\n\n\n/**\n * A cache of the Long representations of small integer values.\n * @type {!Object}\n * @private\n */\nKotlin.Long.IntCache_ = {};\n\n\n/**\n * Returns a Long representing the given (32-bit) integer value.\n * @param {number} value The 32-bit integer in question.\n * @return {!Kotlin.Long} The corresponding Long value.\n */\nKotlin.Long.fromInt = function(value) {\n if (-128 <= value && value < 128) {\n   var cachedObj = Kotlin.Long.IntCache_[value];\n   if (cachedObj) {\n   return cachedObj;\n   }\n }\n var obj = new Kotlin.Long(value | 0, value < 0 ? -1 : 0);\n if (-128 <= value && value < 128) {\n   Kotlin.Long.IntCache_[value] = obj;\n }\n return obj;\n};\n\n\n/**\n * Converts this number value to `Long`.\n * The fractional part, if any, is rounded down towards zero.\n * Returns zero if this `Double` value is `NaN`, `Long.MIN_VALUE` if it's less than `Long.MIN_VALUE`,\n * `Long.MAX_VALUE` if it's bigger than `Long.MAX_VALUE`.\n * @param {number} value The number in question.\n * @return {!Kotlin.Long} The corresponding Long value.\n */\nKotlin.Long.fromNumber =

function(value) {\n if (isNaN(value)) {\n   return Kotlin.Long.ZERO;\n } else if (value <= -Kotlin.Long.TWO_PWR_63_DBL_) {\n   return Kotlin.Long.MIN_VALUE;\n } else if (value + 1 >= Kotlin.Long.TWO_PWR_63_DBL_) {\n   return Kotlin.Long.MAX_VALUE;\n } else if (value < 0) {\n   return Kotlin.Long.fromNumber(-value).negate();\n } else {\n   return new Kotlin.Long(\n   (value % Kotlin.Long.TWO_PWR_32_DBL_) | 0,\n   (value / Kotlin.Long.TWO_PWR_32_DBL_) | 0);\n }\n};\n\n\n/**\n * Returns a Long representing the 64-bit integer that comes by concatenating\n * the given high and low bits. Each is assumed to use 32 bits.\n * @param {number} lowBits The low 32-bits.\n * @param {number} highBits The high 32-bits.\n * @return {!Kotlin.Long} The corresponding Long value.\n */\nKotlin.Long.fromBits = function(lowBits, highBits) {\n return new Kotlin.Long(lowBits, highBits);\n};\n\n\n/**\n * Returns a Long representation of the given string, written using the given\n * radix.\n * @param {string} str The textual representation of the Long.\n * @param {number=} opt_radix The radix in which the text is written.\n * @return {!Kotlin.Long} The corresponding Long value.\n */\nKotlin.Long.fromString = function(str, opt_radix) {\n if (str.length == 0) {\n   throw Error('number format error: empty string');\n }\n\n var radix = opt_radix || 10;\n if (radix < 2 || 36 < radix) {\n   throw Error('radix out of range: ' + radix);\n }\n\n if (str.charAt(0) == '-') {\n   return Kotlin.Long.fromString(str.substring(1), radix).negate();\n } else if (str.indexOf('-') >= 0) {\n   throw Error('number format error: interior \"-\" character: ' + str);\n }\n\n // Do several (8) digits each time through the loop, so as to\n // minimize the calls to the very expensive emulated div.\n var radixToPower = Kotlin.Long.fromNumber(Math.pow(radix, 8));\n\n var result = Kotlin.Long.ZERO;\n for (var i = 0; i < str.length; i += 8) {\n   var size = Math.min(8, str.length - i);\n   var value = parseInt(str.substring(i, i + size), radix);\n   if (size < 8) {\n   var power = Kotlin.Long.fromNumber(Math.pow(radix, size));\n   result = result.multiply(power).add(Kotlin.Long.fromNumber(value));\n   } else {\n   result = result.multiply(radixToPower);\n   result = result.add(Kotlin.Long.fromNumber(value));\n   }\n }\n return result;\n};\n\n\n// NOTE: the compiler should inline these constant values below and then remove\n// these variables, so there should be no runtime penalty for these.\n\n\n/**\n * Number used repeated below in calculations. This must appear before the\n * first call to any from* function below.\n * @type {number}\n * @private\n */\nKotlin.Long.TWO_PWR_16_DBL_ = 1 << 16;\n\n\n/**\n * @type {number}\n * @private\n */\nKotlin.Long.TWO_PWR_24_DBL_ = 1 << 24;\n\n\n/**\n * @type {number}\n * @private\n */\nKotlin.Long.TWO_PWR_32_DBL_ =\n   Kotlin.Long.TWO_PWR_16_DBL_ * Kotlin.Long.TWO_PWR_16_DBL_;\n\n\n/**\n * @type {number}\n * @private\n */\nKotlin.Long.TWO_PWR_31_DBL_ =\n   Kotlin.Long.TWO_PWR_32_DBL_ / 2;\n\n\n/**\n * @type {number}\n * @private\n */\nKotlin.Long.TWO_PWR_48_DBL_ =\n   Kotlin.Long.TWO_PWR_32_DBL_ * Kotlin.Long.TWO_PWR_16_DBL_;\n\n\n/**\n * @type {number}\n * @private\n */\nKotlin.Long.TWO_PWR_64_DBL_ =\n   Kotlin.Long.TWO_PWR_32_DBL_ * Kotlin.Long.TWO_PWR_32_DBL_;\n\n\n/**\n * @type {number}\n * @private\n */\nKotlin.Long.TWO_PWR_63_DBL_ =\n   Kotlin.Long.TWO_PWR_64_DBL_ / 2;\n\n\n/** @type {!Kotlin.Long} */\nKotlin.Long.ZERO = Kotlin.Long.fromInt(0);\n\n\n/** @type {!Kotlin.Long} */\nKotlin.Long.ONE = Kotlin.Long.fromInt(1);\n\n\n/** @type {!Kotlin.Long} */\nKotlin.Long.NEG_ONE = Kotlin.Long.fromInt(-1);\n\n\n/** @type {!Kotlin.Long} */\nKotlin.Long.MAX_VALUE =\n   Kotlin.Long.fromBits(0xFFFFFFFF | 0, 0x7FFFFFFF | 0);\n\n\n/** @type {!Kotlin.Long} */\nKotlin.Long.MIN_VALUE = Kotlin.Long.fromBits(0, 0x80000000 | 0);\n\n\n/**\n * @type {!Kotlin.Long}\n * @private\n */\nKotlin.Long.TWO_PWR_24_ = Kotlin.Long.fromInt(1 << 24);\n\n\n/** @return {number} The value, assuming it is a 32-bit integer. */\nKotlin.Long.prototype.toInt = function() {\n return this.low_;\n};\n\n\n/** @return {number} The closest floating-point representation to this value. */\nKotlin.Long.prototype.toNumber = function() {\n return this.high_ * Kotlin.Long.TWO_PWR_32_DBL_ +\n   this.getLowBitsUnsigned();\n};\n\n/** @return {number} The 32-bit hashCode of this value. */\nKotlin.Long.prototype.hashCode = function() {\n return this.high_ ^ this.low_;\n};\n\n/**\n * @param {number=} opt_radix The radix in which the text should be written.\n * @return {string} The textual representation of this value.\n * @override\n */\nKotlin.Long.prototype.toString = function(opt_radix) {\n var radix = opt_radix ||

```
10;\n  if (radix < 2 || 36 < radix) {\n    throw Error('radix out of range: ' + radix);\n  }\n\n  if (this.isZero()) {\n
return '0';\n  }\n\n  if (this.isNegative()) {\n    if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {\n      // We need to
change the Long value before it can be negated, so we remove\n      // the bottom-most digit in this base and then
recurse to do the rest.\n      var radixLong = Kotlin.Long.fromNumber(radix);\n      var div = this.div(radixLong);\n
 var rem = div.multiply(radixLong).subtract(this);\n      return div.toString(radix) + rem.toInt().toString(radix);\n    }
else {\n      return '-' + this.negate().toString(radix);\n    }\n  }\n\n  // Do several (6) digits each time through the loop,
so as to\n  // minimize the calls to the very expensive emulated div.\n  var radixToPower =
Kotlin.Long.fromNumber(Math.pow(radix, 6));\n\n  var rem = this;\n  var result = '';\n  while (true) {\n    var
remDiv = rem.div(radixToPower);\n    var intval = rem.subtract(remDiv.multiply(radixToPower)).toInt();\n    var
digits = intval.toString(radix);\n\n    rem = remDiv;\n    if (rem.isZero()) {\n      return digits + result;\n    } else {\n
 while (digits.length < 6) {\n        digits = '0' + digits;\n      }\n      result = '' + digits + result;\n    }\n  }\n};\n\n\n/**
@return {number} The high 32-bits as a signed value. */\nKotlin.Long.prototype.getHighBits = function() {\n
return this.high_;\n};\n\n\n/** @return {number} The low 32-bits as a signed value.
*/\nKotlin.Long.prototype.getLowBits = function() {\n  return this.low_;\n};\n\n\n/** @return {number} The low
32-bits as an unsigned value. */\nKotlin.Long.prototype.getLowBitsUnsigned = function() {\n  return (this.low_ >=
0) ?\n      this.low_ : Kotlin.Long.TWO_PWR_32_DBL_ + this.low_;\n};\n\n\n/**\n * @return {number} Returns
the number of bits needed to represent the absolute\n *     value of this Long.\n
*/\nKotlin.Long.prototype.getNumBitsAbs = function() {\n  if (this.isNegative()) {\n    if
(this.equalsLong(Kotlin.Long.MIN_VALUE)) {\n      return 64;\n    } else {\n      return
this.negate().getNumBitsAbs();\n    }\n  } else {\n    var val = this.high_ != 0 ? this.high_ : this.low_;\n    for (var bit
= 31; bit > 0; bit--) {\n      if ((val & (1 << bit)) != 0) {\n        break;\n      }\n    }\n    return this.high_ != 0 ? bit + 33
: bit + 1;\n  }\n};\n\n\n/** @return {boolean} Whether this value is zero. */\nKotlin.Long.prototype.isZero =
function() {\n  return this.high_ == 0 && this.low_ == 0;\n};\n\n\n/** @return {boolean} Whether this value is
negative. */\nKotlin.Long.prototype.isNegative = function() {\n  return this.high_ < 0;\n};\n\n\n/** @return
{boolean} Whether this value is odd. */\nKotlin.Long.prototype.isOdd = function() {\n  return (this.low_ & 1) ==
1;\n};\n\n\n/**\n * @param {Kotlin.Long} other Long to compare against.\n * @return {boolean} Whether this
Long equals the other.\n */\nKotlin.Long.prototype.equalsLong = function(other) {\n  return (this.high_ ==
other.high_) && (this.low_ == other.low_);\n};\n\n\n/**\n * @param {Kotlin.Long} other Long to compare
against.\n * @return {boolean} Whether this Long does not equal the other.\n
*/\nKotlin.Long.prototype.notEqualsLong = function(other) {\n  return (this.high_ != other.high_) || (this.low_ !=
other.low_);\n};\n\n\n/**\n * @param {Kotlin.Long} other Long to compare against.\n * @return {boolean}
Whether this Long is less than the other.\n */\nKotlin.Long.prototype.lessThan = function(other) {\n  return
this.compare(other) < 0;\n};\n\n\n/**\n * @param {Kotlin.Long} other Long to compare against.\n * @return
{boolean} Whether this Long is less than or equal to the other.\n */\nKotlin.Long.prototype.lessThanOrEqual =
function(other) {\n  return this.compare(other) <= 0;\n};\n\n\n/**\n * @param {Kotlin.Long} other Long to
compare against.\n * @return {boolean} Whether this Long is greater than the other.\n
*/\nKotlin.Long.prototype.greaterThan = function(other) {\n  return this.compare(other) > 0;\n};\n\n\n/**\n *
@param {Kotlin.Long} other Long to compare against.\n * @return {boolean} Whether this Long is greater than or
equal to the other.\n */\nKotlin.Long.prototype.greaterThanOrEqual = function(other) {\n  return
this.compare(other) >= 0;\n};\n\n\n/**\n * Compares this Long with the given one.\n * @param {Kotlin.Long}
other Long to compare against.\n * @return {number} 0 if they are the same, 1 if the this is greater, and -1\n *     if
the given one is greater.\n */\nKotlin.Long.prototype.compare = function(other) {\n  if (this.equalsLong(other)) {\n
return 0;\n  }\n\n  var thisNeg = this.isNegative();\n  var otherNeg = other.isNegative();\n  if (thisNeg &&
!otherNeg) {\n    return -1;\n  }\n  if (!thisNeg && otherNeg) {\n    return 1;\n  }\n\n  // at this point, the signs are the
same, so subtraction will not overflow\n  if (this.subtract(other).isNegative()) {\n    return -1;\n  } else {\n    return
1;\n  }\n};\n\n\n/** @return {!Kotlin.Long} The negation of this value. */\nKotlin.Long.prototype.negate =
function() {\n  if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {\n    return Kotlin.Long.MIN_VALUE;\n  } else
{\n    return this.not().add(Kotlin.Long.ONE);\n  }\n};\n\n\n/**\n * Returns the sum of this and the given Long.\n *
```

@param {Kotlin.Long} other Long to add to this one.\n * @return {!Kotlin.Long} The sum of this and the given Long.\n */\nKotlin.Long.prototype.add = function(other) {\n // Divide each number into 4 chunks of 16 bits, and then sum the chunks.\n\n var a48 = this.high_ >>> 16;\n var a32 = this.high_ & 0xFFFF;\n var a16 = this.low_ >>> 16;\n var a00 = this.low_ & 0xFFFF;\n\n var b48 = other.high_ >>> 16;\n var b32 = other.high_ & 0xFFFF;\n var b16 = other.low_ >>> 16;\n var b00 = other.low_ & 0xFFFF;\n\n var c48 = 0, c32 = 0, c16 = 0, c00 = 0;\n c00 += a00 + b00;\n c16 += c00 >>> 16;\n c00 &= 0xFFFF;\n c16 += a16 + b16;\n c32 += c16 >>> 16;\n c16 &= 0xFFFF;\n c32 += a32 + b32;\n c48 += c32 >>> 16;\n c32 &= 0xFFFF;\n c48 += a48 + b48;\n c48 &= 0xFFFF;\n return Kotlin.Long.fromBits((c16 << 16) | c00, (c48 << 16) | c32);\n};\n\n\n/**\n * Returns the difference of this and the given Long.\n * @param {Kotlin.Long} other Long to subtract from this.\n * @return {!Kotlin.Long} The difference of this and the given Long.\n */\nKotlin.Long.prototype.subtract = function(other) {\n return this.add(other.negate());\n};\n\n\n/**\n * Returns the product of this and the given long.\n * @param {Kotlin.Long} other Long to multiply with this.\n * @return {!Kotlin.Long} The product of this and the other.\n */\nKotlin.Long.prototype.multiply = function(other) {\n if (this.isZero()) {\n   return Kotlin.Long.ZERO;\n } else if (other.isZero()) {\n   return Kotlin.Long.ZERO;\n }\n\n if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {\n   return other.isOdd() ? Kotlin.Long.MIN_VALUE : Kotlin.Long.ZERO;\n } else if (other.equalsLong(Kotlin.Long.MIN_VALUE)) {\n   return this.isOdd() ? Kotlin.Long.MIN_VALUE : Kotlin.Long.ZERO;\n }\n\n if (this.isNegative()) {\n   if (other.isNegative()) {\n   return this.negate().multiply(other.negate());\n   } else {\n   return this.negate().multiply(other).negate();\n   }\n } else if (other.isNegative()) {\n   return this.multiply(other.negate()).negate();\n }\n\n // If both longs are small, use float multiplication\n if (this.lessThan(Kotlin.Long.TWO_PWR_24_) &&\n   other.lessThan(Kotlin.Long.TWO_PWR_24_)) {\n   return Kotlin.Long.fromNumber(this.toNumber() * other.toNumber());\n }\n\n // Divide each long into 4 chunks of 16 bits, and then add up 4x4 products.\n // We can skip products that would overflow.\n\n var a48 = this.high_ >>> 16;\n var a32 = this.high_ & 0xFFFF;\n var a16 = this.low_ >>> 16;\n var a00 = this.low_ & 0xFFFF;\n\n var b48 = other.high_ >>> 16;\n var b32 = other.high_ & 0xFFFF;\n var b16 = other.low_ >>> 16;\n var b00 = other.low_ & 0xFFFF;\n\n var c48 = 0, c32 = 0, c16 = 0, c00 = 0;\n c00 += a00 * b00;\n c16 += c00 >>> 16;\n c00 &= 0xFFFF;\n c16 += a16 * b00;\n c32 += c16 >>> 16;\n c16 &= 0xFFFF;\n c16 += a00 * b16;\n c32 += c16 >>> 16;\n c16 &= 0xFFFF;\n c32 += a32 * b00;\n c48 += c32 >>> 16;\n c32 &= 0xFFFF;\n c32 += a16 * b16;\n c48 += c32 >>> 16;\n c32 &= 0xFFFF;\n c32 += a00 * b32;\n c48 += c32 >>> 16;\n c32 &= 0xFFFF;\n c48 += a48 * b00 + a32 * b16 + a16 * b32 + a00 * b48;\n c48 &= 0xFFFF;\n return Kotlin.Long.fromBits((c16 << 16) | c00, (c48 << 16) | c32);\n};\n\n\n/**\n * Returns this Long divided by the given one.\n * @param {Kotlin.Long} other Long by which to divide.\n * @return {!Kotlin.Long} This Long divided by the given one.\n */\nKotlin.Long.prototype.div = function(other) {\n if (other.isZero()) {\n   throw Error('division by zero');\n } else if (this.isZero()) {\n   return Kotlin.Long.ZERO;\n }\n\n if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {\n   if (other.equalsLong(Kotlin.Long.ONE) ||\n   other.equalsLong(Kotlin.Long.NEG_ONE)) {\n   return Kotlin.Long.MIN_VALUE; // recall that -MIN_VALUE == MIN_VALUE\n   } else if (other.equalsLong(Kotlin.Long.MIN_VALUE)) {\n   return Kotlin.Long.ONE;\n   } else {\n   // At this point, we have |other| >= 2, so |this/other| < |MIN_VALUE|.\n   var halfThis = this.shiftRight(1);\n   var approx = halfThis.div(other).shiftLeft(1);\n   if (approx.equalsLong(Kotlin.Long.ZERO)) {\n   return other.isNegative() ? Kotlin.Long.ONE : Kotlin.Long.NEG_ONE;\n   } else {\n   var rem = this.subtract(other.multiply(approx));\n   var result = approx.add(rem.div(other));\n   return result;\n   }\n   }\n } else if (other.equalsLong(Kotlin.Long.MIN_VALUE)) {\n   return Kotlin.Long.ZERO;\n }\n\n if (this.isNegative()) {\n   if (other.isNegative()) {\n   return this.negate().div(other.negate());\n   } else {\n   return this.negate().div(other).negate();\n   }\n } else if (other.isNegative()) {\n   return this.div(other.negate()).negate();\n }\n\n // Repeat the following until the remainder is less than other: find a\n // floating-point that approximates remainder / other *from below*, add this\n // into the result, and subtract it from the remainder. It is critical that\n // the approximate value is less than or equal to the real value so that the\n // remainder never becomes negative.\n var res = Kotlin.Long.ZERO;\n var rem = this;\n while

(rem.greaterThanOrEqual(other)) {\n // Approximate the result of division. This may be a little greater or\n // smaller than the actual value.\n var approx = Math.max(1, Math.floor(rem.toNumber() / other.toNumber()));\n\n // We will tweak the approximate result by changing it in the 48-th digit or\n // the smallest non-fractional digit, whichever is larger.\n var log2 = Math.ceil(Math.log(approx) / Math.LN2);\n var delta = (log2 <= 48) ? 1 : Math.pow(2, log2 - 48);\n\n // Decrease the approximation until it is smaller than the remainder. Note\n // that if it is too large, the product overflows and is negative.\n var approxRes = Kotlin.Long.fromNumber(approx);\n var approxRem = approxRes.multiply(other);\n while (approxRem.isNegative() || approxRem.greaterThan(rem)) {\n approx -= delta;\n approxRes = Kotlin.Long.fromNumber(approx);\n approxRem = approxRes.multiply(other);\n }\n\n // We know the answer can't be zero... and actually, zero would cause\n // infinite recursion since we would make no progress.\n if (approxRes.isZero()) {\n approxRes = Kotlin.Long.ONE;\n }\n\n res = res.add(approxRes);\n rem = rem.subtract(approxRem);\n }\n return res;\n};\n\n\n/**\n * Returns this Long modulo the given one.\n * @param {Kotlin.Long} other Long by which to mod.\n * @return {!Kotlin.Long} This Long modulo the given one.\n */\nKotlin.Long.prototype.modulo = function(other) {\n return this.subtract(this.div(other).multiply(other));\n};\n\n\n/** @return {!Kotlin.Long} The bitwise-NOT of this value. */\nKotlin.Long.prototype.not = function() {\n return Kotlin.Long.fromBits(~this.low_, ~this.high_);\n};\n\n\n/**\n * Returns the bitwise-AND of this Long and the given one.\n * @param {Kotlin.Long} other The Long with which to AND.\n * @return {!Kotlin.Long} The bitwise-AND of this and the other.\n */\nKotlin.Long.prototype.and = function(other) {\n return Kotlin.Long.fromBits(this.low_ & other.low_,\n this.high_ & other.high_);\n};\n\n\n/**\n * Returns the bitwise-OR of this Long and the given one.\n * @param {Kotlin.Long} other The Long with which to OR.\n * @return {!Kotlin.Long} The bitwise-OR of this and the other.\n */\nKotlin.Long.prototype.or = function(other) {\n return Kotlin.Long.fromBits(this.low_ | other.low_,\n this.high_ | other.high_);\n};\n\n\n/**\n * Returns the bitwise-XOR of this Long and the given one.\n * @param {Kotlin.Long} other The Long with which to XOR.\n * @return {!Kotlin.Long} The bitwise-XOR of this and the other.\n */\nKotlin.Long.prototype.xor = function(other) {\n return Kotlin.Long.fromBits(this.low_ ^ other.low_,\n this.high_ ^ other.high_);\n};\n\n\n/**\n * Returns this Long with bits shifted to the left by the given amount.\n * @param {number} numBits The number of bits by which to shift.\n * @return {!Kotlin.Long} This shifted to the left by the given amount.\n */\nKotlin.Long.prototype.shiftLeft = function(numBits) {\n numBits &= 63;\n if (numBits == 0) {\n return this;\n } else {\n var low = this.low_;\n if (numBits < 32) {\n var high = this.high_;\n return Kotlin.Long.fromBits(\n low << numBits,\n (high << numBits) | (low >>> (32 - numBits)));\n } else {\n return Kotlin.Long.fromBits(0, low << (numBits - 32));\n }\n }\n};\n\n\n/**\n * Returns this Long with bits shifted to the right by the given amount.\n * @param {number} numBits The number of bits by which to shift.\n * @return {!Kotlin.Long} This shifted to the right by the given amount.\n */\nKotlin.Long.prototype.shiftRight = function(numBits) {\n numBits &= 63;\n if (numBits == 0) {\n return this;\n } else {\n var high = this.high_;\n if (numBits < 32) {\n var low = this.low_;\n return Kotlin.Long.fromBits(\n (low >>> numBits) | (high << (32 - numBits)),\n high >> numBits);\n } else {\n return Kotlin.Long.fromBits(\n high >> (numBits - 32),\n high >= 0 ? 0 : -1);\n }\n }\n};\n\n\n/**\n * Returns this Long with bits shifted to the right by the given amount, with\n * zeros placed into the new leading bits.\n * @param {number} numBits The number of bits by which to shift.\n * @return {!Kotlin.Long} This shifted to the right by the given amount, with\n * zeros placed into the new leading bits.\n */\nKotlin.Long.prototype.shiftRightUnsigned = function(numBits) {\n numBits &= 63;\n if (numBits == 0) {\n return this;\n } else {\n var high = this.high_;\n if (numBits < 32) {\n var low = this.low_;\n return Kotlin.Long.fromBits(\n (low >>> numBits) | (high << (32 - numBits)),\n high >>> numBits);\n } else if (numBits == 32) {\n return Kotlin.Long.fromBits(high, 0);\n } else {\n return Kotlin.Long.fromBits(high >>> (numBits - 32), 0);\n }\n }\n};\n\n// Support for Kotlin\nKotlin.Long.prototype.equals = function (other) {\n return other instanceof Kotlin.Long && this.equalsLong(other);\n};\n\nKotlin.Long.prototype.compareTo_11rb$ = Kotlin.Long.prototype.compare;\n\nKotlin.Long.prototype.inc = function() {\n return this.add(Kotlin.Long.ONE);\n};\n\nKotlin.Long.prototype.dec = function() {\n return

this.add(Kotlin.Long.NEG_ONE);\n};\n\nKotlin.Long.prototype.valueOf = function() {\n return this.toNumber();\n};\n\nKotlin.Long.prototype.unaryPlus = function() {\n return this;\n};\n\nKotlin.Long.prototype.unaryMinus = Kotlin.Long.prototype.negate;\nKotlin.Long.prototype.inv = Kotlin.Long.prototype.not;\n\nKotlin.Long.prototype.rangeTo = function (other) {\n return new Kotlin.kotlin.ranges.LongRange(this, other);\n};","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n/**\n * @param {string} id\n * @param {Object} declaration\n */\nKotlin.defineModule = function (id, declaration) {\n};\n\nKotlin.defineInlineFunction = function(tag, fun) {\n return fun;\n};\n\nKotlin.wrapFunction = function(fun) {\n var f = function() {\n   f = fun();\n   return f.apply(this, arguments);\n };\n return function() {\n   return f.apply(this, arguments);\n };\n};\n\nKotlin.isTypeOf = function(type) {\n return function (object) {\n   return typeof object === type;\n }\n};\n\nKotlin.isInstanceOf = function (klass) {\n return function (object) {\n   return Kotlin.isType(object, klass);\n }\n};\n\nKotlin.orNull = function (fn) {\n return function (object) {\n   return object == null || fn(object);\n }\n};\n\nKotlin.andPredicate = function (a, b) {\n return function (object) {\n   return a(object) && b(object);\n }\n};\n\nKotlin.kotlinModuleMetadata = function (abiVersion, moduleName, data) {\n};\n\nKotlin.suspendCall = function(value) {\n return value;\n};\n\nKotlin.coroutineResult = function(qualifier) {\n throwMarkerError();\n};\n\nKotlin.coroutineController = function(qualifier) {\n throwMarkerError();\n};\n\nKotlin.coroutineReceiver = function(qualifier) {\n throwMarkerError();\n};\n\nKotlin.setCoroutineResult = function(value, qualifier) {\n throwMarkerError();\n};\n\nKotlin.getReifiedTypeParameterKType = function(typeParameter) {\n throwMarkerError();\n};\n\nfunction throwMarkerError() {\n throw new Error(\n   \"This marker function should never been called. \" +\n   \"Looks like compiler did not eliminate it properly. \" +\n   \"Please, report an issue if you caught this exception.\");\n}\n\nKotlin.getFunctionById = function(id, defaultValue) {\n return function() {\n   return defaultValue;\n }\n};","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\nKotlin.compareTo = function (a, b) {\n var typeA = typeof a;\n if (typeA === \"number\") {\n   if (typeof b === \"number\") {\n   return Kotlin.doubleCompareTo(a, b);\n   }\n   return Kotlin.primitiveCompareTo(a, b);\n }\n if (typeA === \"string\" || typeA === \"boolean\") {\n   return Kotlin.primitiveCompareTo(a, b);\n }\n return a.compareTo_11rb$(b);\n};\n\nKotlin.primitiveCompareTo = function (a, b) {\n return a < b ? -1 : a > b ? 1 : 0;\n};\n\nKotlin.doubleCompareTo = function (a, b) {\n if (a < b) return -1;\n if (a > b) return 1;\n\n if (a === b) {\n   if (a !== 0) return 0;\n\n   var ia = 1 / a;\n   return ia === 1 / b ? 0 : (ia < 0 ? -1 : 1);\n }\n\n return a !== a ? (b !== b ? 0 : 1) : -1\n};\n\nKotlin.charInc = function (value) {\n return Kotlin.toChar(value+1);\n};\n\nKotlin.charDec = function (value) {\n return Kotlin.toChar(value-1);\n};\n\nKotlin.imul = Math.imul || imul;\n\nKotlin.imulEmulated = imul;\n\nfunction imul(a, b) {\n return ((a & 0xffff0000) * (b & 0xffff) + (a & 0xffff) * (b | 0)) | 0;\n}\n\n(function() {\n var buf = new ArrayBuffer(8);\n var bufFloat64 = new Float64Array(buf);\n var bufFloat32 = new Float32Array(buf);\n var bufInt32 = new Int32Array(buf);\n var lowIndex = 0;\n var highIndex = 1;\n\n bufFloat64[0] = -1; // bff00000_00000000\n if (bufInt32[lowIndex] !== 0) {\n   lowIndex = 1;\n   highIndex = 0;\n }\n\n Kotlin.doubleToBits = function(value) {\n   return Kotlin.doubleToRawBits(isNaN(value) ? NaN : value);\n };\n\n Kotlin.doubleToRawBits = function(value) {\n   bufFloat64[0] = value;\n   return Kotlin.Long.fromBits(bufInt32[lowIndex], bufInt32[highIndex]);\n };\n\n Kotlin.doubleFromBits = function(value) {\n   bufInt32[lowIndex] = value.low_;\n   bufInt32[highIndex] = value.high_;\n   return bufFloat64[0];\n };\n\n Kotlin.floatToBits = function(value) {\n   return Kotlin.floatToRawBits(isNaN(value) ? NaN : value);\n };\n\n Kotlin.floatToRawBits = function(value) {\n   bufFloat32[0] = value;\n   return bufInt32[0];\n };\n\n Kotlin.floatFromBits = function(value) {\n   bufInt32[0] = value;\n   return bufFloat32[0];\n };\n\n // returns zero value for number with positive sign bit and non-zero value for number with negative sign bit.\n Kotlin.doubleSignBit = function(value) {\n   bufFloat64[0] = value;\n   return

bufInt32[highIndex] & 0x80000000;\n };\n\n Kotlin.numberHashCode = function(obj) {\n if ((obj | 0) === obj) {\n return obj | 0;\n }\n else {\n bufFloat64[0] = obj;\n return (bufInt32[highIndex] * 31 | 0) + bufInt32[lowIndex] | 0;\n }\n }\n})();\n\nKotlin.ensureNotNull = function(x) {\n return x != null ? x : Kotlin.throwNPE();\n};\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\nif (typeof String.prototype.startsWith === \"undefined\") {\n Object.defineProperty(String.prototype, \"startsWith\", {\n value: function (searchString, position) {\n position = position || 0;\n return this.lastIndexOf(searchString, position) === position;\n }\n });\n}\nif (typeof String.prototype.endsWith === \"undefined\") {\n Object.defineProperty(String.prototype, \"endsWith\", {\n value: function (searchString, position) {\n var subjectString = this.toString();\n if (position === undefined || position > subjectString.length) {\n position = subjectString.length;\n }\n position -= searchString.length;\n var lastIndex = subjectString.indexOf(searchString, position);\n return lastIndex !== -1 && lastIndex === position;\n }\n });\n}\n// ES6 Math polyfills\nif (typeof Math.sign === \"undefined\") {\n Math.sign = function(x) {\n x = +x; // convert to a number\n if (x === 0 || isNaN(x)) {\n return Number(x);\n }\n return x > 0 ? 1 : -1;\n };\n}\nif (typeof Math.trunc === \"undefined\") {\n Math.trunc = function(x) {\n if (isNaN(x)) {\n return NaN;\n }\n if (x > 0) {\n return Math.floor(x);\n }\n return Math.ceil(x);\n };\n}\n\n(function() {\n var epsilon = 2.220446049250313E-16;\n var taylor_2_bound = Math.sqrt(epsilon);\n var taylor_n_bound = Math.sqrt(taylor_2_bound);\n var upper_taylor_2_bound = 1/taylor_2_bound;\n var upper_taylor_n_bound = 1/taylor_n_bound;\n\n if (typeof Math.sinh === \"undefined\") {\n Math.sinh = function(x) {\n if (Math.abs(x) < taylor_n_bound) {\n var result = x;\n if (Math.abs(x) > taylor_2_bound) {\n result += (x * x * x) / 6;\n }\n return result;\n } else {\n var y = Math.exp(x);\n var y1 = 1 / y;\n if (!isFinite(y)) return Math.exp(x - Math.LN2);\n if (!isFinite(y1)) return -Math.exp(-x - Math.LN2);\n return (y - y1) / 2;\n }\n };\n }\n if (typeof Math.cosh === \"undefined\") {\n Math.cosh = function(x) {\n var y = Math.exp(x);\n var y1 = 1 / y;\n if (!isFinite(y) || !isFinite(y1)) return Math.exp(Math.abs(x) - Math.LN2);\n return (y + y1) / 2;\n };\n }\n\n if (typeof Math.tanh === \"undefined\") {\n Math.tanh = function(x){\n if (Math.abs(x) < taylor_n_bound) {\n var result = x;\n if (Math.abs(x) > taylor_2_bound) {\n result -= (x * x * x) / 3;\n }\n return result;\n }\n else {\n var a = Math.exp(+x), b = Math.exp(-x);\n return a === Infinity ? 1 : b === Infinity ? -1 : (a - b) / (a + b);\n }\n };\n }\n\n // Inverse hyperbolic function implementations derived from boost special math functions,\n // Copyright Eric Ford & Hubert Holin 2001.\n\n if (typeof Math.asinh === \"undefined\") {\n var asinh = function(x) {\n if (x >= +taylor_n_bound)\n {\n if (x > upper_taylor_n_bound)\n {\n if (x > upper_taylor_2_bound)\n {\n // approximation by laurent series in 1/x at 0+ order from -1 to 0\n return Math.log(x) + Math.LN2;\n }\n else\n {\n // approximation by laurent series in 1/x at 0+ order from -1 to 1\n return Math.log(x * 2 + (1 / (x * 2)));\n }\n }\n else\n {\n return Math.log(x + Math.sqrt(x * x + 1));\n }\n }\n else if (x <= -taylor_n_bound)\n {\n return -asinh(-x);\n }\n else\n {\n // approximation by taylor series in x at 0 up to order 2\n var result = x;\n if (Math.abs(x) >= taylor_2_bound)\n {\n var x3 = x * x * x;\n // approximation by taylor series in x at 0 up to order 4\n result -= x3 / 6;\n }\n return result;\n }\n };\n Math.asinh = asinh;\n }\n if (typeof Math.acosh === \"undefined\") {\n Math.acosh = function(x) {\n if (x < 1)\n {\n return NaN;\n }\n else if (x - 1 >= taylor_n_bound)\n {\n if (x > upper_taylor_2_bound)\n {\n // approximation by laurent series in 1/x at 0+ order from -1 to 0\n return Math.log(x) + Math.LN2;\n }\n else\n {\n return Math.log(x + Math.sqrt(x * x - 1));\n }\n }\n else\n {\n var y = Math.sqrt(x - 1);\n // approximation by taylor series in y at 0 up to order 2\n var result = y;\n if (y >= taylor_2_bound)\n {\n var y3 = y *

```
y * y;\n                // approximation by taylor series in y at 0 up to order 4\n                result -= y3 / 12;\n
}\n\n            return Math.sqrt(2) * result;\n        }\n    };\n  }\n  if (typeof Math.atanh === \"undefined\")
{\n    Math.atanh = function(x) {\n        if (Math.abs(x) < taylor_n_bound) {\n            var result = x;\n
if (Math.abs(x) > taylor_2_bound) {\n                result += (x * x * x) / 3;\n            }\n            return result;\n
    }\n        return Math.log((1 + x) / (1 - x)) / 2;\n    };\n  }\n  if (typeof Math.log1p === \"undefined\") {\n
  Math.log1p = function(x) {\n        if (Math.abs(x) < taylor_n_bound) {\n            var x2 = x * x;\n
var x3 = x2 * x;\n            var x4 = x3 * x;\n            // approximation by taylor series in x at 0 up to order 4\n
    return (-x4 / 4 + x3 / 3 - x2 / 2 + x);\n        }\n        return Math.log(x + 1);\n    };\n  }\n  if (typeof
Math.expm1 === \"undefined\") {\n    Math.expm1 = function(x) {\n        if (Math.abs(x) < taylor_n_bound)
{\n            var x2 = x * x;\n            var x3 = x2 * x;\n            var x4 = x3 * x;\n            // approximation by
taylor series in x at 0 up to order 4\n            return (x4 / 24 + x3 / 6 + x2 / 2 + x);\n        }\n        return
Math.exp(x) - 1;\n    };\n  }\n})();\nif (typeof Math.hypot === \"undefined\") {\n  Math.hypot = function() {\n
  var y = 0;\n    var length = arguments.length;\n\n    for (var i = 0; i < length; i++) {\n        if (arguments[i]
=== Infinity || arguments[i] === -Infinity) {\n            return Infinity;\n        }\n        y += arguments[i] *
arguments[i];\n    }\n    return Math.sqrt(y);\n  };\n}\nif (typeof Math.log10 === \"undefined\") {\n
Math.log10 = function(x) {\n    return Math.log(x) * Math.LOG10E;\n  };\n}\nif (typeof Math.log2 ===
\"undefined\") {\n  Math.log2 = function(x) {\n    return Math.log(x) * Math.LOG2E;\n  };\n}\nif (typeof
Math.clz32 === \"undefined\") {\n  Math.clz32 = (function(log, LN2) {\n    return function(x) {\n        var
asUint = x >>> 0;\n        if (asUint === 0) {\n            return 32;\n        }\n        return 31 - (log(asUint) /
LN2 | 0) | 0; // the \"| 0\" acts like math.floor\n    };\n  })(Math.log, Math.LN2);\n}\n\n// For HtmlUnit and
PhantomJs\nif (typeof ArrayBuffer.isView === \"undefined\") {\n  ArrayBuffer.isView = function(a) {\n
return a != null && a.__proto__ != null && a.__proto__.__proto__ === Int8Array.prototype.__proto__;\n
};\n}\n\nif (typeof Array.prototype.fill === \"undefined\") {\n  // Polyfill from https://developer.mozilla.org/en-
US/docs/Web/JavaScript/Reference/Global_Objects/Array/fill#Polyfill\n  Object.defineProperty(Array.prototype,
'fill', {\n    value: function (value) {\n\n        // Steps 1-2.\n        if (this == null) {\n            throw new
TypeError('this is null or not defined');\n        }\n\n        var O = Object(this);\n\n        // Steps 3-5.\n
var len = O.length >>> 0;\n\n        // Steps 6-7.\n        var start = arguments[1];\n        var relativeStart = start
>> 0;\n\n        // Step 8.\n        var k = relativeStart < 0 ?\n            Math.max(len + relativeStart, 0) :\n
    Math.min(relativeStart, len);\n\n        // Steps 9-10.\n        var end = arguments[2];\n        var
relativeEnd = end === undefined ?\n                len : end >> 0;\n\n        // Step 11.\n        var finalValue
= relativeEnd < 0 ?\n                Math.max(len + relativeEnd, 0) :\n                Math.min(relativeEnd,
len);\n\n        // Step 12.\n        while (k < finalValue) {\n            O[k] = value;\n            k++;\n        }\n\n
    // Step 13.\n        return O;\n    }\n  });\n}\n\n(function() {\n  function normalizeOffset(offset, length)
{\n    if (offset < 0) return Math.max(0, offset + length);\n    return Math.min(offset, length);\n  }\n  function
typedArraySlice(begin, end) {\n    if (typeof end === \"undefined\") {\n        end = this.length;\n    }\n
begin = normalizeOffset(begin || 0, this.length);\n    end = Math.max(begin, normalizeOffset(end, this.length));\n
  return new this.constructor(this.subarray(begin, end));\n  }\n\n  var arrays = [Int8Array, Int16Array,
Uint16Array, Int32Array, Float32Array, Float64Array];\n  for (var i = 0; i < arrays.length; ++i) {\n    var
TypedArray = arrays[i];\n    if (typeof TypedArray.prototype.fill === \"undefined\") {\n
Object.defineProperty(TypedArray.prototype, 'fill', {\n            value: Array.prototype.fill\n        });\n    }\n
  if (typeof TypedArray.prototype.slice === \"undefined\") {\n        Object.defineProperty(TypedArray.prototype,
'slice', {\n            value: typedArraySlice\n        });\n    }\n  }\n\n  // Patch apply to work with TypedArrays
if needed.\n  try {\n    (function() {}).apply(null, new Int32Array(0))\n  } catch (e) {\n    var apply =
Function.prototype.apply;\n    Object.defineProperty(Function.prototype, 'apply', {\n        value: function(self,
array) {\n            return apply.call(this, self, [].slice.call(array));\n        }\n    });\n  }\n\n\n  // Patch map to
work with TypedArrays if needed.\n  for (var i = 0; i < arrays.length; ++i) {\n    var TypedArray = arrays[i];\n
  if (typeof TypedArray.prototype.map === \"undefined\") {\n        Object.defineProperty(TypedArray.prototype,
'map', {\n            value: function(callback, self) {\n                return [].slice.call(this).map(callback, self);\n
```

}\n });\n }\n }\n\n // Patch sort to work with TypedArrays if needed.\n // TODO: consider to remove following function and replace it with `Kotlin.doubleCompareTo` (see misc.js)\n var totalOrderComparator = function (a, b) {\n if (a < b) return -1;\n if (a > b) return 1;\n\n if (a === b) {\n if (a !== 0) return 0;\n\n var ia = 1 / a;\n return ia === 1 / b ? 0 : (ia < 0 ? -1 : 1);\n }\n\n return a !== a ? (b !== b ? 0 : 1) : -1\n };\n\n for (var i = 0; i < arrays.length; ++i) {\n var TypedArray = arrays[i];\n if (typeof TypedArray.prototype.sort === \"undefined\") {\n Object.defineProperty(TypedArray.prototype, 'sort', {\n value: function(compareFunction) {\n return Array.prototype.sort.call(this, compareFunction || totalOrderComparator);\n }\n });\n }\n }\n})();\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\nKotlin.Kind = {\n CLASS: \"class\",\n INTERFACE: \"interface\",\n OBJECT: \"object\"\n};\n\nKotlin.callGetter = function (thisObject, klass, propertyName) {\n var propertyDescriptor = Object.getOwnPropertyDescriptor(klass, propertyName);\n if (propertyDescriptor != null && propertyDescriptor.get != null) {\n return propertyDescriptor.get.call(thisObject);\n }\n\n propertyDescriptor = Object.getOwnPropertyDescriptor(thisObject, propertyName);\n if (propertyDescriptor != null && \"value\" in propertyDescriptor) {\n return thisObject[propertyName];\n }\n\n return Kotlin.callGetter(thisObject, Object.getPrototypeOf(klass), propertyName);\n};\n\nKotlin.callSetter = function (thisObject, klass, propertyName, value) {\n var propertyDescriptor = Object.getOwnPropertyDescriptor(klass, propertyName);\n if (propertyDescriptor != null && propertyDescriptor.set != null) {\n propertyDescriptor.set.call(thisObject, value);\n return;\n }\n\n propertyDescriptor = Object.getOwnPropertyDescriptor(thisObject, propertyName);\n if (propertyDescriptor != null && \"value\" in propertyDescriptor) {\n thisObject[propertyName] = value;\n return\n }\n\n Kotlin.callSetter(thisObject, Object.getPrototypeOf(klass), propertyName, value);\n};\n\nfunction isInheritanceFromInterface(ctor, iface) {\n if (ctor === iface) return true;\n\n var metadata = ctor.$metadata$;\n if (metadata != null) {\n var interfaces = metadata.interfaces;\n for (var i = 0; i < interfaces.length; i++) {\n if (isInheritanceFromInterface(interfaces[i], iface)) {\n return true;\n }\n }\n }\n\n var superPrototype = ctor.prototype != null ? Object.getPrototypeOf(ctor.prototype) : null;\n var superConstructor = superPrototype != null ? superPrototype.constructor : null;\n return superConstructor != null && isInheritanceFromInterface(superConstructor, iface);\n}\n\n/**\n *\n * @param {*} object\n * @param {Function|Object} klass\n * @returns {Boolean}\n */\nKotlin.isType = function (object, klass) {\n if (klass === Object) {\n switch (typeof object) {\n case \"string\":\n case \"number\":\n case \"boolean\":\n case \"function\":\n return true;\n default:\n return object instanceof Object;\n }\n }\n\n if (object == null || klass == null || (typeof object !== 'object' && typeof object !== 'function')) {\n return false;\n }\n\n if (typeof klass === \"function\" && object instanceof klass) {\n return true;\n }\n\n var proto = Object.getPrototypeOf(klass);\n var constructor = proto != null ? proto.constructor : null;\n if (constructor != null && \"$metadata$\" in constructor) {\n var metadata = constructor.$metadata$;\n if (metadata.kind === Kotlin.Kind.OBJECT) {\n return object === klass;\n }\n }\n\n var klassMetadata = klass.$metadata$;\n\n // In WebKit (JavaScriptCore) for some interfaces from DOM typeof returns \"object\", nevertheless they can be used in RHS of instanceof\n if (klassMetadata == null) {\n return object instanceof klass;\n }\n\n if (klassMetadata.kind === Kotlin.Kind.INTERFACE && object.constructor != null) {\n return isInheritanceFromInterface(object.constructor, klass);\n }\n\n return false;\n};\n\nKotlin.isNumber = function (a) {\n return typeof a == \"number\" || a instanceof Kotlin.Long;\n};\n\nKotlin.isChar = function (value) {\n return value instanceof Kotlin.BoxedChar\n};\n\nKotlin.isComparable = function (value) {\n var type = typeof value;\n\n return type === \"string\" ||\n type === \"boolean\" ||\n Kotlin.isNumber(value) ||\n Kotlin.isType(value, Kotlin.kotlin.Comparable);\n};\n\nKotlin.isCharSequence = function (value) {\n return typeof value === \"string\" || Kotlin.isType(value, Kotlin.kotlin.CharSequence);\n};\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be

found in the license/LICENSE.txt file.\n */\n\n// a package is omitted to get declarations directly under the module\n\n@PublishedApi\nexternal internal fun <T> Array(size: Int): Array<T>\n\n@JsName(\"newArray\")\nfun <T> newArray(size: Int, initValue: T) = fillArrayVal(Array<T>(size), initValue)\n\n@JsName(\"newArrayF\")\ninline fun <T> arrayWithFun(size: Int, init: (Int) -> T) = fillArrayFun(Array<T>(size), init)\n\n@JsName(\"fillArray\")\ninline fun <T> fillArrayFun(array: Array<T>, init: (Int) -> T): Array<T> {\n for (i in 0..array.size - 1) {\n array[i] = init(i)\n }\n return array\n}\n\n@JsName(\"booleanArray\")\nfun booleanArray(size: Int, init: dynamic): Array<Boolean> {\n val result: dynamic = Array<Boolean>(size)\n result.`$type$` = \"BooleanArray\"\n return when (init) {\n null, true -> fillArrayVal(result, false)\n false -> result\n else -> fillArrayFun<Boolean>(result, init)\n }\n}\n\n@JsName(\"booleanArrayF\")\ninline fun booleanArrayWithFun(size: Int, init: (Int) -> Boolean): Array<Boolean> = fillArrayFun(booleanArray(size, false), init)\n\n@JsName(\"charArray\")\n@Suppress(\"UNUSED_PARAMETER\")\nfun charArray(size: Int, init: dynamic): Array<Char> {\n val result = js(\"new Uint16Array(size)\")\n result.`$type$` = \"CharArray\"\n return when (init) {\n null, true, false -> result // For consistency\n else -> fillArrayFun<Char>(result, init)\n }\n}\n\n@JsName(\"charArrayF\")\ninline fun charArrayWithFun(size: Int, init: (Int) -> Char): Array<Char> {\n val array = charArray(size, null)\n for (i in 0..array.size - 1) {\n @Suppress(\"UNUSED_VARIABLE\") // used in js block\n val value = init(i)\n js(\"array[i] = value;\")\n }\n return array\n}\n\n@JsName(\"untypedCharArrayF\")\ninline fun untypedCharArrayWithFun(size: Int, init: (Int) -> Char): Array<Char> {\n val array = Array<Char>(size)\n for (i in 0..array.size - 1) {\n @Suppress(\"UNUSED_VARIABLE\") // used in js block\n val value = init(i)\n js(\"array[i] = value;\")\n }\n return array\n}\n\n@JsName(\"longArray\")\nfun longArray(size: Int, init: dynamic): Array<Long> {\n val result: dynamic = Array<Long>(size)\n result.`$type$` = \"LongArray\"\n return when (init) {\n null, true -> fillArrayVal(result, 0L)\n false -> result\n else -> fillArrayFun<Long>(result, init)\n }\n}\n\n@JsName(\"longArrayF\")\ninline fun longArrayWithFun(size: Int, init: (Int) -> Long): Array<Long> = fillArrayFun(longArray(size, false), init)\n\nprivate fun <T> fillArrayVal(array: Array<T>, initValue: T): Array<T> {\n for (i in 0..array.size - 1) {\n array[i] = initValue\n }\n return array\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\npublic class Enum<T : Enum<T>> : Comparable<Enum<T>> {\n @JsName(\"name$\") private var _name: String = \"\"\n @JsName(\"ordinal$\") private var _ordinal: Int = 0\n\n val name: String\n get() = _name\n\n val ordinal: Int\n get() = _ordinal\n\n override fun compareTo(other: Enum<T>) = ordinal.compareTo(other.ordinal)\n\n override fun equals(other: Any?) = this === other\n\n override fun hashCode(): Int = js(\"Kotlin.identityHashCode\")(this)\n\n override fun toString() = name\n\n companion object\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js.internal\n\n@JsName(\"DoubleCompanionObject\")\ninternal object DoubleCompanionObject {\n @JsName(\"MIN_VALUE\")\n const val MIN_VALUE: Double = 4.9E-324\n\n @JsName(\"MAX_VALUE\")\n const val MAX_VALUE: Double = 1.7976931348623157E308\n\n @JsName(\"POSITIVE_INFINITY\")\n @Suppress(\"DIVISION_BY_ZERO\")\n const val POSITIVE_INFINITY: Double = 1.0 / 0.0\n\n @JsName(\"NEGATIVE_INFINITY\")\n @Suppress(\"DIVISION_BY_ZERO\")\n const val NEGATIVE_INFINITY: Double = -1.0 / 0.0\n\n @JsName(\"NaN\")\n @Suppress(\"DIVISION_BY_ZERO\")\n const val NaN: Double = -(0.0 / 0.0)\n\n @JsName(\"SIZE_BYTES\")\n const val SIZE_BYTES = 8\n\n @JsName(\"SIZE_BITS\")\n const val SIZE_BITS = 64\n}\n\n@JsName(\"FloatCompanionObject\")\ninternal object FloatCompanionObject {\n @JsName(\"MIN_VALUE\")\n const val MIN_VALUE: Float = 1.4E-45F\n\n @JsName(\"MAX_VALUE\")\n const val MAX_VALUE: Float = 3.4028235E38F\n\n @JsName(\"POSITIVE_INFINITY\")\n @Suppress(\"DIVISION_BY_ZERO\")\n const val POSITIVE_INFINITY: Float = 1.0F / 0.0F\n\n @JsName(\"NEGATIVE_INFINITY\")\n @Suppress(\"DIVISION_BY_ZERO\")\n const val

NEGATIVE_INFINITY: Float = -1.0F / 0.0F\n\n   @JsName(\"NaN\")\n
@Suppress(\"DIVISION_BY_ZERO\")\n   const val NaN: Float = -(0.0F / 0.0F)\n\n
@JsName(\"SIZE_BYTES\")\n   const val SIZE_BYTES = 4\n\n   @JsName(\"SIZE_BITS\")\n   const val
SIZE_BITS = 32\n}\n\n@JsName(\"IntCompanionObject\")\ninternal object IntCompanionObject {\n
@JsName(\"MIN_VALUE\")\n   val MIN_VALUE: Int = -2147483647 - 1\n\n   @JsName(\"MAX_VALUE\")\n
val MAX_VALUE: Int = 2147483647\n\n   @JsName(\"SIZE_BYTES\")\n   const val SIZE_BYTES = 4\n\n
@JsName(\"SIZE_BITS\")\n   const val SIZE_BITS = 32\n}\n\n@JsName(\"LongCompanionObject\")\ninternal
object LongCompanionObject {\n   @JsName(\"MIN_VALUE\")\n   val MIN_VALUE: Long =
js(\"Kotlin.Long.MIN_VALUE\")\n\n   @JsName(\"MAX_VALUE\")\n   val MAX_VALUE: Long =
js(\"Kotlin.Long.MAX_VALUE\")\n\n   @JsName(\"SIZE_BYTES\")\n   const val SIZE_BYTES = 8\n\n
@JsName(\"SIZE_BITS\")\n   const val SIZE_BITS = 64\n}\n\n@JsName(\"ShortCompanionObject\")\ninternal
object ShortCompanionObject {\n   @JsName(\"MIN_VALUE\")\n   val MIN_VALUE: Short = -32768\n\n
@JsName(\"MAX_VALUE\")\n   val MAX_VALUE: Short = 32767\n\n   @JsName(\"SIZE_BYTES\")\n   const
val SIZE_BYTES = 2\n\n   @JsName(\"SIZE_BITS\")\n   const val SIZE_BITS =
16\n}\n\n@JsName(\"ByteCompanionObject\")\ninternal object ByteCompanionObject {\n
@JsName(\"MIN_VALUE\")\n   val MIN_VALUE: Byte = -128\n\n   @JsName(\"MAX_VALUE\")\n   val
MAX_VALUE: Byte = 127\n\n   @JsName(\"SIZE_BYTES\")\n   const val SIZE_BYTES = 1\n\n
@JsName(\"SIZE_BITS\")\n   const val SIZE_BITS = 8\n}\n\n@JsName(\"CharCompanionObject\")\ninternal
object CharCompanionObject {\n   @JsName(\"MIN_VALUE\")\n   public const val MIN_VALUE: Char =
'\\u0000'\n\n   @JsName(\"MAX_VALUE\")\n   public const val MAX_VALUE: Char = '\\uFFFF'\n\n
@JsName(\"MIN_HIGH_SURROGATE\")\n   public const val MIN_HIGH_SURROGATE: Char = '\\uD800'\n\n
@JsName(\"MAX_HIGH_SURROGATE\")\n   public const val MAX_HIGH_SURROGATE: Char =
'\\uDBFF'\n\n   @JsName(\"MIN_LOW_SURROGATE\")\n   public const val MIN_LOW_SURROGATE: Char =
'\\uDC00'\n\n   @JsName(\"MAX_LOW_SURROGATE\")\n   public const val MAX_LOW_SURROGATE: Char
= '\\uDFFF'\n\n   @JsName(\"MIN_SURROGATE\")\n   public const val MIN_SURROGATE: Char =
MIN_HIGH_SURROGATE\n\n   @JsName(\"MAX_SURROGATE\")\n   public const val MAX_SURROGATE:
Char = MAX_LOW_SURROGATE\n\n   @JsName(\"SIZE_BYTES\")\n   const val SIZE_BYTES = 2\n\n
@JsName(\"SIZE_BITS\")\n   const val SIZE_BITS = 16\n}\n\ninternal object StringCompanionObject
{}\n\ninternal object BooleanCompanionObject {}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin
Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\n
*/\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"ArraysKt\")\n\npackage
kotlin.collections\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\nimport
kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the
size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior
is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> Array<out T>.component1(): T
{\n   return get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1,
throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component1(): Byte {\n   return
get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an
[IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component1(): Short {\n   return
get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an
[IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component1(): Int {\n   return
get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an
[IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n

*/\n@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component1(): Long {\n   return get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component1(): Float {\n   return get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component1(): Double {\n   return get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component1(): Boolean {\n   return get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component1(): Char {\n   return get(0)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> Array<out T>.component2(): T {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component2(): Byte {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component2(): Short {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component2(): Int {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component2(): Long {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component2(): Float {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component2(): Double {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component2(): Boolean {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component2(): Char {\n   return get(1)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> Array<out T>.component3(): T {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component3(): Byte {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n

*/\n@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component3(): Short {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component3(): Int {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component3(): Long {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component3(): Float {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component3(): Double {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component3(): Boolean {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component3(): Char {\n   return get(2)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> Array<out T>.component4(): T {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component4(): Byte {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component4(): Short {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component4(): Int {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component4(): Long {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component4(): Float {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component4(): Double {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component4(): Boolean {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component4(): Char {\n   return get(3)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n

*/\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> Array<out T>.component5(): T {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component5(): Byte {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component5(): Short {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component5(): Int {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component5(): Long {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component5(): Float {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component5(): Double {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component5(): Boolean {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component5(): Char {\n   return get(4)\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n */\npublic operator fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.contains(element: T): Boolean {\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n */\npublic operator fun ByteArray.contains(element: Byte): Boolean {\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n */\npublic operator fun ShortArray.contains(element: Short): Boolean {\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n */\npublic operator fun IntArray.contains(element: Int): Boolean {\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n */\npublic operator fun LongArray.contains(element: Long): Boolean {\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n */\n@Deprecated(\"The function has unclear behavior when searching for NaN or zero values and will be removed soon. Use 'any { it == element }' instead to continue using this behavior, or '.asList().contains(element: T)' to get the same search behavior as in a list.\", ReplaceWith(\"any { it == element }\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\n@Suppress(\"DEPRECATION_ERROR\")\npublic operator fun FloatArray.contains(element: Float): Boolean {\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n */\n@Deprecated(\"The function has unclear behavior when searching for NaN or zero values and will be removed soon. Use 'any { it == element }' instead to continue using this behavior, or '.asList().contains(element: T)' to get the same search behavior as in a list.\", ReplaceWith(\"any { it == element }\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\n@Suppress(\"DEPRECATION_ERROR\")\npublic operator fun DoubleArray.contains(element: Double): Boolean {\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n */\npublic operator fun BooleanArray.contains(element: Boolean): Boolean {\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n */\npublic operator fun CharArray.contains(element: Char): Boolean {\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns an element

at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic expect fun <T> Array<out T>.elementAt(index: Int): T\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic expect fun ByteArray.elementAt(index: Int): Byte\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic expect fun ShortArray.elementAt(index: Int): Short\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic expect fun IntArray.elementAt(index: Int): Int\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic expect fun LongArray.elementAt(index: Int): Long\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic expect fun FloatArray.elementAt(index: Int): Float\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic expect fun DoubleArray.elementAt(index: Int): Double\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic expect fun BooleanArray.elementAt(index: Int): Boolean\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic expect fun CharArray.elementAt(index: Int): Char\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Byte): Byte {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Short): Short {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Int): Int {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Long): Long {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Float): Float {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample

samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Double): Double {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Char): Char {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.elementAtOrNull(index: Int): T? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.elementAtOrNull(index: Int): Byte? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.elementAtOrNull(index: Int): Short? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.elementAtOrNull(index: Int): Int? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.elementAtOrNull(index: Int): Long? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.elementAtOrNull(index: Int): Float? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.elementAtOrNull(index: Int): Double? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.elementAtOrNull(index: Int): Boolean? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.elementAtOrNull(index: Int): Char? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.find(predicate: (T) -> Boolean): T? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.find(predicate: (Byte) -> Boolean): Byte? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.find(predicate: (Short) -> Boolean): Short? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample

samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.find(predicate: (Int) -> Boolean): Int? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.find(predicate: (Long) -> Boolean): Long? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.find(predicate: (Float) -> Boolean): Float? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.find(predicate: (Double) -> Boolean): Double? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.find(predicate: (Boolean) -> Boolean): Boolean? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.find(predicate: (Char) -> Boolean): Char? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.findLast(predicate: (T) -> Boolean): T? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.findLast(predicate: (Byte) -> Boolean): Byte? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.findLast(predicate: (Short) -> Boolean): Short? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.findLast(predicate: (Int) -> Boolean): Int? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.findLast(predicate: (Long) -> Boolean): Long? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.findLast(predicate: (Float) -> Boolean): Float? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.findLast(predicate: (Double) -> Boolean): Double? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.findLast(predicate: (Boolean) -> Boolean): Boolean? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.findLast(predicate: (Char) -> Boolean): Char? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun <T> Array<out T>.first(): T {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun ByteArray.first(): Byte {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is

empty.\")\n   return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun ShortArray.first(): Short {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun IntArray.first(): Int {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun LongArray.first(): Long {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun FloatArray.first(): Float {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun DoubleArray.first(): Double {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun BooleanArray.first(): Boolean {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun CharArray.first(): Char {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return this[0]\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun <T> Array<out T>.first(predicate: (T) -> Boolean): T {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun ByteArray.first(predicate: (Byte) -> Boolean): Byte {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun ShortArray.first(predicate: (Short) -> Boolean): Short {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun IntArray.first(predicate: (Int) -> Boolean): Int {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun LongArray.first(predicate: (Long) -> Boolean): Long {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun FloatArray.first(predicate: (Float) -> Boolean): Float {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun DoubleArray.first(predicate: (Double) -> Boolean): Double {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun BooleanArray.first(predicate: (Boolean) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun CharArray.first(predicate: (Char) -> Boolean): Char {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first non-null value produced by [transform] function being applied to elements of this array in iteration order,\n * or

throws [NoSuchElementException] if no non-null value was produced.\n * \n * @sample samples.collections.Collections.Transformations.firstNotNullOf\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Any> Array<out T>.firstNotNullOf(transform: (T) -> R?): R {\n return firstNotNullOfOrNull(transform) ?: throw NoSuchElementException(\"No element of the array was transformed to a non-null value.\")\n}\n\n/**\n * Returns the first non-null value produced by [transform] function being applied to elements of this array in iteration order,\n * or `null` if no non-null value was produced.\n * \n * @sample samples.collections.Collections.Transformations.firstNotNullOf\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Any> Array<out T>.firstNotNullOfOrNull(transform: (T) -> R?): R? {\n for (element in this) {\n val result = transform(element)\n if (result != null) {\n return result\n }\n }\n return null\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun <T> Array<out T>.firstOrNull(): T? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun ByteArray.firstOrNull(): Byte? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun ShortArray.firstOrNull(): Short? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun IntArray.firstOrNull(): Int? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun LongArray.firstOrNull(): Long? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun FloatArray.firstOrNull(): Float? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun DoubleArray.firstOrNull(): Double? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun BooleanArray.firstOrNull(): Boolean? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun CharArray.firstOrNull(): Char? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun <T> Array<out T>.firstOrNull(predicate: (T) -> Boolean): T? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun ByteArray.firstOrNull(predicate: (Byte) -> Boolean): Byte? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun ShortArray.firstOrNull(predicate: (Short) -> Boolean): Short? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun IntArray.firstOrNull(predicate: (Int) -> Boolean): Int? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun LongArray.firstOrNull(predicate: (Long) -> Boolean): Long? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun FloatArray.firstOrNull(predicate: (Float) -> Boolean): Float? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun DoubleArray.firstOrNull(predicate: (Double) -> Boolean): Double? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun BooleanArray.firstOrNull(predicate: (Boolean) -> Boolean): Boolean? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun CharArray.firstOrNull(predicate: (Char) -> Boolean): Char? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n

*/\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.getOrElse(index: Int, defaultValue: (Int) -> Byte): Byte {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.getOrElse(index: Int, defaultValue: (Int) -> Short): Short {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.getOrElse(index: Int, defaultValue: (Int) -> Int): Int {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.getOrElse(index: Int, defaultValue: (Int) -> Long): Long {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.getOrElse(index: Int, defaultValue: (Int) -> Float): Float {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.getOrElse(index: Int, defaultValue: (Int) -> Double): Double {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.getOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n */\npublic fun <T> Array<out T>.getOrNull(index: Int): T? {\n   return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n */\npublic fun ByteArray.getOrNull(index: Int): Byte? {\n return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n */\npublic fun ShortArray.getOrNull(index: Int): Short? {\n return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n */\npublic fun IntArray.getOrNull(index: Int): Int? {\n return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n */\npublic fun LongArray.getOrNull(index: Int): Long? {\n return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n */\npublic fun FloatArray.getOrNull(index: Int): Float? {\n return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n */\npublic fun DoubleArray.getOrNull(index: Int): Double?

{\n   return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n */\npublic fun BooleanArray.getOrNull(index: Int): Boolean? {\n   return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n */\npublic fun CharArray.getOrNull(index: Int): Char? {\n   return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.indexOf(element: T): Int {\n   if (element == null) {\n   for (index in indices) {\n   if (this[index] == null) {\n   return index\n   }\n   }\n   } else {\n   for (index in indices) {\n   if (element == this[index]) {\n   return index\n   }\n   }\n   }\n   return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun ByteArray.indexOf(element: Byte): Int {\n   for (index in indices) {\n   if (element == this[index]) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun ShortArray.indexOf(element: Short): Int {\n   for (index in indices) {\n   if (element == this[index]) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun IntArray.indexOf(element: Int): Int {\n   for (index in indices) {\n   if (element == this[index]) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun LongArray.indexOf(element: Long): Int {\n   for (index in indices) {\n   if (element == this[index]) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\n@Deprecated(\"The function has unclear behavior when searching for NaN or zero values and will be removed soon. Use 'indexOfFirst { it == element }' instead to continue using this behavior, or '.asList().indexOf(element: T)' to get the same search behavior as in a list.\", ReplaceWith(\"indexOfFirst { it == element }\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\npublic fun FloatArray.indexOf(element: Float): Int {\n   for (index in indices) {\n   if (element == this[index]) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\n@Deprecated(\"The function has unclear behavior when searching for NaN or zero values and will be removed soon. Use 'indexOfFirst { it == element }' instead to continue using this behavior, or '.asList().indexOf(element: T)' to get the same search behavior as in a list.\", ReplaceWith(\"indexOfFirst { it == element }\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\npublic fun DoubleArray.indexOf(element: Double): Int {\n   for (index in indices) {\n   if (element == this[index]) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun BooleanArray.indexOf(element: Boolean): Int {\n   for (index in indices) {\n   if (element == this[index]) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun CharArray.indexOf(element: Char): Int {\n   for (index in indices) {\n   if (element == this[index]) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun <T> Array<out T>.indexOfFirst(predicate: (T) -> Boolean): Int {\n   for (index in indices) {\n   if (predicate(this[index])) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun ByteArray.indexOfFirst(predicate: (Byte) -> Boolean): Int {\n   for (index in indices) {\n   if (predicate(this[index])) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun ShortArray.indexOfFirst(predicate: (Short) -> Boolean): Int {\n   for (index in indices) {\n   if (predicate(this[index])) {\n   return index\n   }\n   }\n   return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun IntArray.indexOfFirst(predicate: (Int) -> Boolean): Int {\n   for (index in indices) {\n   if (predicate(this[index]))

{\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun LongArray.indexOfFirst(predicate: (Long) -> Boolean): Int {\n for (index in indices) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun FloatArray.indexOfFirst(predicate: (Float) -> Boolean): Int {\n for (index in indices) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun DoubleArray.indexOfFirst(predicate: (Double) -> Boolean): Int {\n for (index in indices) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun BooleanArray.indexOfFirst(predicate: (Boolean) -> Boolean): Int {\n for (index in indices) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun CharArray.indexOfFirst(predicate: (Char) -> Boolean): Int {\n for (index in indices) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun <T> Array<out T>.indexOfLast(predicate: (T) -> Boolean): Int {\n for (index in indices.reversed()) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun ByteArray.indexOfLast(predicate: (Byte) -> Boolean): Int {\n for (index in indices.reversed()) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun ShortArray.indexOfLast(predicate: (Short) -> Boolean): Int {\n for (index in indices.reversed()) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun IntArray.indexOfLast(predicate: (Int) -> Boolean): Int {\n for (index in indices.reversed()) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun LongArray.indexOfLast(predicate: (Long) -> Boolean): Int {\n for (index in indices.reversed()) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun FloatArray.indexOfLast(predicate: (Float) -> Boolean): Int {\n for (index in indices.reversed()) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun DoubleArray.indexOfLast(predicate: (Double) -> Boolean): Int {\n for (index in indices.reversed()) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun BooleanArray.indexOfLast(predicate: (Boolean) -> Boolean): Int {\n for (index in indices.reversed()) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun CharArray.indexOfLast(predicate: (Char) -> Boolean): Int {\n for (index in indices.reversed()) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun <T> Array<out T>.last(): T {\n if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n return this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun ByteArray.last(): Byte {\n if (isEmpty())\n

throw NoSuchElementException(\"Array is empty.\")\n   return this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun ShortArray.last(): Short {\n   if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n   return this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun IntArray.last(): Int {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun LongArray.last(): Long {\n   if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n   return this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun FloatArray.last(): Float {\n   if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n   return this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun DoubleArray.last(): Double {\n   if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n   return this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun BooleanArray.last(): Boolean {\n   if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n   return this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun CharArray.last(): Char {\n   if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n   return this[lastIndex]\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun <T> Array<out T>.last(predicate: (T) -> Boolean): T {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun ByteArray.last(predicate: (Byte) -> Boolean): Byte {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun ShortArray.last(predicate: (Short) -> Boolean): Short {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun IntArray.last(predicate: (Int) -> Boolean): Int {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun LongArray.last(predicate: (Long) -> Boolean): Long {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample

samples.collections.Collections.Elements.last\n */\npublic inline fun FloatArray.last(predicate: (Float) -> Boolean): Float {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun DoubleArray.last(predicate: (Double) -> Boolean): Double {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun BooleanArray.last(predicate: (Boolean) -> Boolean): Boolean {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun CharArray.last(predicate: (Char) -> Boolean): Char {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\npublic fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.lastIndexOf(element: T): Int {\n if (element == null) {\n for (index in indices.reversed()) {\n if (this[index] == null) {\n return index\n }\n }\n } else {\n for (index in indices.reversed()) {\n if (element == this[index]) {\n return index\n }\n }\n }\n return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\npublic fun ByteArray.lastIndexOf(element: Byte): Int {\n for (index in indices.reversed()) {\n if (element == this[index]) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\npublic fun ShortArray.lastIndexOf(element: Short): Int {\n for (index in indices.reversed()) {\n if (element == this[index]) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\npublic fun IntArray.lastIndexOf(element: Int): Int {\n for (index in indices.reversed()) {\n if (element == this[index]) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\npublic fun LongArray.lastIndexOf(element: Long): Int {\n for (index in indices.reversed()) {\n if (element == this[index]) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\n@Deprecated(\"The function has unclear behavior when searching for NaN or zero values and will be removed soon. Use 'indexOfLast { it == element }' instead to continue using this behavior, or '.asList().lastIndexOf(element: T)' to get the same search behavior as in a list.\", ReplaceWith(\"indexOfLast { it == element }\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\npublic fun FloatArray.lastIndexOf(element: Float): Int {\n for (index in indices.reversed()) {\n if (element == this[index]) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\n@Deprecated(\"The function has unclear behavior when searching for NaN or zero values and will be removed soon. Use 'indexOfLast { it == element }' instead to continue using this behavior, or '.asList().lastIndexOf(element: T)' to get the same search behavior as in a list.\", ReplaceWith(\"indexOfLast { it == element }\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\npublic fun DoubleArray.lastIndexOf(element: Double): Int {\n for (index in indices.reversed()) {\n if (element == this[index]) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\npublic fun BooleanArray.lastIndexOf(element: Boolean): Int {\n for (index in indices.reversed()) {\n if (element == this[index]) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\npublic fun CharArray.lastIndexOf(element: Char): Int {\n for (index in

indices.reversed()) {\n if (element == this[index]) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun <T> Array<out T>.lastOrNull(): T? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun ByteArray.lastOrNull(): Byte? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun ShortArray.lastOrNull(): Short? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun IntArray.lastOrNull(): Int? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun LongArray.lastOrNull(): Long? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun FloatArray.lastOrNull(): Float? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun DoubleArray.lastOrNull(): Double? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun BooleanArray.lastOrNull(): Boolean? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun CharArray.lastOrNull(): Char? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun <T> Array<out T>.lastOrNull(predicate: (T) -> Boolean): T? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun ByteArray.lastOrNull(predicate: (Byte) -> Boolean): Byte? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun ShortArray.lastOrNull(predicate: (Short) -> Boolean): Short? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun IntArray.lastOrNull(predicate: (Int) -> Boolean): Int? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun LongArray.lastOrNull(predicate: (Long) -> Boolean): Long? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun FloatArray.lastOrNull(predicate: (Float) -> Boolean): Float? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun DoubleArray.lastOrNull(predicate: (Double) -> Boolean): Double? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns the last element matching the given

[predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun BooleanArray.lastOrNull(predicate: (Boolean) -> Boolean): Boolean? {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   return null\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun CharArray.lastOrNull(predicate: (Char) -> Boolean): Char? {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   return null\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.random(): T {\n return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.random(): Byte {\n   return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.random(): Short {\n   return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.random(): Int {\n   return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.random(): Long {\n   return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.random(): Float {\n   return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.random(): Double {\n return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.random(): Boolean {\n   return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.random(): Char {\n   return random(Random)\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun <T> Array<out T>.random(random: Random): T {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun ByteArray.random(random: Random): Byte {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun ShortArray.random(random: Random): Short {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun IntArray.random(random: Random): Int {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun LongArray.random(random: Random): Long {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns a random

element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun FloatArray.random(random: Random): Float {\n if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun DoubleArray.random(random: Random): Double {\n if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun BooleanArray.random(random: Random): Boolean {\n if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun CharArray.random(random: Random): Char {\n if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.randomOrNull(): T? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.randomOrNull(): Byte? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.randomOrNull(): Short? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.randomOrNull(): Int? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.randomOrNull(): Long? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.randomOrNull(): Float? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.randomOrNull(): Double? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.randomOrNull(): Boolean? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.randomOrNull(): Char? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun <T> Array<out T>.randomOrNull(random: Random): T? {\n if (isEmpty())\n return null\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun ByteArray.randomOrNull(random: Random): Byte? {\n if (isEmpty())\n return null\n return

get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun ShortArray.randomOrNull(random: Random): Short? {\n if (isEmpty())\n return null\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun IntArray.randomOrNull(random: Random): Int? {\n if (isEmpty())\n return null\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun LongArray.randomOrNull(random: Random): Long? {\n if (isEmpty())\n return null\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun FloatArray.randomOrNull(random: Random): Float? {\n if (isEmpty())\n return null\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun DoubleArray.randomOrNull(random: Random): Double? {\n if (isEmpty())\n return null\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun BooleanArray.randomOrNull(random: Random): Boolean? {\n if (isEmpty())\n return null\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun CharArray.randomOrNull(random: Random): Char? {\n if (isEmpty())\n return null\n return get(random.nextInt(size))\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun <T> Array<out T>.single(): T {\n return when (size) {\n 0 -> throw NoSuchElementException(\"Array is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"Array has more than one element.\")\n }\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun ByteArray.single(): Byte {\n return when (size) {\n 0 -> throw NoSuchElementException(\"Array is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"Array has more than one element.\")\n }\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun ShortArray.single(): Short {\n return when (size) {\n 0 -> throw NoSuchElementException(\"Array is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"Array has more than one element.\")\n }\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun IntArray.single(): Int {\n return when (size) {\n 0 -> throw NoSuchElementException(\"Array is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"Array has more than one element.\")\n }\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun LongArray.single(): Long {\n return when (size) {\n 0 -> throw NoSuchElementException(\"Array is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"Array has more than one element.\")\n }\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun FloatArray.single(): Float {\n return when (size) {\n 0 -> throw NoSuchElementException(\"Array is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"Array has more than one element.\")\n

}\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun DoubleArray.single(): Double {\n return when (size) {\n 0 -> throw NoSuchElementException(\"Array is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"Array has more than one element.\")\n }\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun BooleanArray.single(): Boolean {\n return when (size) {\n 0 -> throw NoSuchElementException(\"Array is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"Array has more than one element.\")\n }\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun CharArray.single(): Char {\n return when (size) {\n 0 -> throw NoSuchElementException(\"Array is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"Array has more than one element.\")\n }\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun <T> Array<out T>.single(predicate: (T) -> Boolean): T {\n var single: T? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as T\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun ByteArray.single(predicate: (Byte) -> Boolean): Byte {\n var single: Byte? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as Byte\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun ShortArray.single(predicate: (Short) -> Boolean): Short {\n var single: Short? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as Short\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun IntArray.single(predicate: (Int) -> Boolean): Int {\n var single: Int? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as Int\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun LongArray.single(predicate: (Long) -> Boolean): Long {\n var single: Long? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as Long\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun FloatArray.single(predicate: (Float) -> Boolean): Float {\n var single: Float? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as Float\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun DoubleArray.single(predicate: (Double) -> Boolean): Double {\n var single:

```
Double? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n            single = element\n            found = true\n        }\n    }\n    if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n    @Suppress(\"UNCHECKED_CAST\")\n    return single as Double\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun BooleanArray.single(predicate: (Boolean) -> Boolean): Boolean {\n    var single: Boolean? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n            single = element\n            found = true\n        }\n    }\n    if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n    @Suppress(\"UNCHECKED_CAST\")\n    return single as Boolean\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun CharArray.single(predicate: (Char) -> Boolean): Char {\n    var single: Char? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n            single = element\n            found = true\n        }\n    }\n    if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n    @Suppress(\"UNCHECKED_CAST\")\n    return single as Char\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\npublic fun <T> Array<out T>.singleOrNull(): T? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\npublic fun ByteArray.singleOrNull(): Byte? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\npublic fun ShortArray.singleOrNull(): Short? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\npublic fun IntArray.singleOrNull(): Int? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\npublic fun LongArray.singleOrNull(): Long? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\npublic fun FloatArray.singleOrNull(): Float? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\npublic fun DoubleArray.singleOrNull(): Double? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\npublic fun BooleanArray.singleOrNull(): Boolean? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\npublic fun CharArray.singleOrNull(): Char? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic inline fun <T> Array<out T>.singleOrNull(predicate: (T) -> Boolean): T? {\n    var single: T? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single = element\n            found = true\n        }\n    }\n    if (!found) return null\n    return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic inline fun ByteArray.singleOrNull(predicate: (Byte) -> Boolean): Byte? {\n    var single: Byte? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single = element\n            found = true\n        }\n    }\n    if (!found) return null\n    return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic inline fun ShortArray.singleOrNull(predicate: (Short) -> Boolean): Short? {\n    var single: Short? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single = element\n            found = true\n        }\n    }\n    if (!found) return null\n    return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic inline fun IntArray.singleOrNull(predicate: (Int) -> Boolean): Int? {\n    var single: Int? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single = element\n            found = true\n        }\n    }\n    if (!found) return null\n    return single\n}\n\n/**\n *
```

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic inline fun LongArray.singleOrNull(predicate: (Long) -> Boolean): Long? {\n var single: Long? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) return null\n single = element\n found = true\n }\n }\n if (!found) return null\n return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic inline fun FloatArray.singleOrNull(predicate: (Float) -> Boolean): Float? {\n var single: Float? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) return null\n single = element\n found = true\n }\n }\n if (!found) return null\n return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic inline fun DoubleArray.singleOrNull(predicate: (Double) -> Boolean): Double? {\n var single: Double? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) return null\n single = element\n found = true\n }\n }\n if (!found) return null\n return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic inline fun BooleanArray.singleOrNull(predicate: (Boolean) -> Boolean): Boolean? {\n var single: Boolean? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) return null\n single = element\n found = true\n }\n }\n if (!found) return null\n return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic inline fun CharArray.singleOrNull(predicate: (Char) -> Boolean): Char? {\n var single: Char? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) return null\n single = element\n found = true\n }\n }\n if (!found) return null\n return single\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun <T> Array<out T>.drop(n: Int): List<T> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun ByteArray.drop(n: Int): List<Byte> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun ShortArray.drop(n: Int): List<Short> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun IntArray.drop(n: Int): List<Int> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun LongArray.drop(n: Int): List<Long> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun FloatArray.drop(n: Int): List<Float> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun DoubleArray.drop(n: Int): List<Double> {\n

require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun BooleanArray.drop(n: Int): List<Boolean> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun CharArray.drop(n: Int): List<Char> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun <T> Array<out T>.dropLast(n: Int): List<T> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun ByteArray.dropLast(n: Int): List<Byte> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun ShortArray.dropLast(n: Int): List<Short> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun IntArray.dropLast(n: Int): List<Int> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun LongArray.dropLast(n: Int): List<Long> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun FloatArray.dropLast(n: Int): List<Float> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun DoubleArray.dropLast(n: Int): List<Double> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun BooleanArray.dropLast(n: Int): List<Boolean> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun CharArray.dropLast(n: Int): List<Char> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun <T> Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {\n   for (index in lastIndex downTo 0) {\n   if

(!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun ByteArray.dropLastWhile(predicate: (Byte) -> Boolean): List<Byte> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun ShortArray.dropLastWhile(predicate: (Short) -> Boolean): List<Short> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun IntArray.dropLastWhile(predicate: (Int) -> Boolean): List<Int> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun LongArray.dropLastWhile(predicate: (Long) -> Boolean): List<Long> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun FloatArray.dropLastWhile(predicate: (Float) -> Boolean): List<Float> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun DoubleArray.dropLastWhile(predicate: (Double) -> Boolean): List<Double> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun BooleanArray.dropLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun CharArray.dropLastWhile(predicate: (Char) -> Boolean): List<Char> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun <T> Array<out T>.dropWhile(predicate: (T) -> Boolean): List<T> {\n var yielding = false\n val list = ArrayList<T>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun ByteArray.dropWhile(predicate: (Byte) -> Boolean): List<Byte> {\n var yielding = false\n val list = ArrayList<Byte>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun ShortArray.dropWhile(predicate: (Short) -> Boolean): List<Short> {\n var yielding = false\n val list = ArrayList<Short>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun

IntArray.dropWhile(predicate: (Int) -> Boolean): List<Int> {\n var yielding = false\n val list = ArrayList<Int>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun LongArray.dropWhile(predicate: (Long) -> Boolean): List<Long> {\n var yielding = false\n val list = ArrayList<Long>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun FloatArray.dropWhile(predicate: (Float) -> Boolean): List<Float> {\n var yielding = false\n val list = ArrayList<Float>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun DoubleArray.dropWhile(predicate: (Double) -> Boolean): List<Double> {\n var yielding = false\n val list = ArrayList<Double>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun BooleanArray.dropWhile(predicate: (Boolean) -> Boolean): List<Boolean> {\n var yielding = false\n val list = ArrayList<Boolean>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun CharArray.dropWhile(predicate: (Char) -> Boolean): List<Char> {\n var yielding = false\n val list = ArrayList<Char>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun <T> Array<out T>.filter(predicate: (T) -> Boolean): List<T> {\n return filterTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun ByteArray.filter(predicate: (Byte) -> Boolean): List<Byte> {\n return filterTo(ArrayList<Byte>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun ShortArray.filter(predicate: (Short) -> Boolean): List<Short> {\n return filterTo(ArrayList<Short>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun IntArray.filter(predicate: (Int) -> Boolean): List<Int> {\n return filterTo(ArrayList<Int>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun LongArray.filter(predicate: (Long) -> Boolean): List<Long> {\n return filterTo(ArrayList<Long>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun FloatArray.filter(predicate: (Float) -> Boolean): List<Float> {\n return filterTo(ArrayList<Float>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun DoubleArray.filter(predicate: (Double) -> Boolean): List<Double> {\n return filterTo(ArrayList<Double>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun BooleanArray.filter(predicate: (Boolean) -> Boolean): List<Boolean> {\n return filterTo(ArrayList<Boolean>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun CharArray.filter(predicate: (Char) ->

Boolean): List<Char> {\n   return filterTo(ArrayList<Char>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun <T> Array<out T>.filterIndexed(predicate: (index: Int, T) -> Boolean): List<T> {\n   return filterIndexedTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun ByteArray.filterIndexed(predicate: (index: Int, Byte) -> Boolean): List<Byte> {\n   return filterIndexedTo(ArrayList<Byte>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun ShortArray.filterIndexed(predicate: (index: Int, Short) -> Boolean): List<Short> {\n   return filterIndexedTo(ArrayList<Short>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun IntArray.filterIndexed(predicate: (index: Int, Int) -> Boolean): List<Int> {\n   return filterIndexedTo(ArrayList<Int>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun LongArray.filterIndexed(predicate: (index: Int, Long) -> Boolean): List<Long> {\n   return filterIndexedTo(ArrayList<Long>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun FloatArray.filterIndexed(predicate: (index: Int, Float) -> Boolean): List<Float> {\n   return filterIndexedTo(ArrayList<Float>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun DoubleArray.filterIndexed(predicate: (index: Int, Double) -> Boolean): List<Double> {\n   return filterIndexedTo(ArrayList<Double>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun BooleanArray.filterIndexed(predicate: (index: Int, Boolean) -> Boolean): List<Boolean> {\n   return filterIndexedTo(ArrayList<Boolean>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun CharArray.filterIndexed(predicate: (index: Int, Char) -> Boolean): List<Char> {\n   return filterIndexedTo(ArrayList<Char>(), predicate)\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <T, C : MutableCollection<in T>> Array<out T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of

predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <C : MutableCollection<in Byte>> ByteArray.filterIndexedTo(destination: C, predicate: (index: Int, Byte) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <C : MutableCollection<in Short>> ShortArray.filterIndexedTo(destination: C, predicate: (index: Int, Short) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <C : MutableCollection<in Int>> IntArray.filterIndexedTo(destination: C, predicate: (index: Int, Int) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <C : MutableCollection<in Long>> LongArray.filterIndexedTo(destination: C, predicate: (index: Int, Long) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <C : MutableCollection<in Float>> FloatArray.filterIndexedTo(destination: C, predicate: (index: Int, Float) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <C : MutableCollection<in Double>> DoubleArray.filterIndexedTo(destination: C, predicate: (index: Int, Double) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <C : MutableCollection<in Boolean>> BooleanArray.filterIndexedTo(destination: C, predicate: (index: Int, Boolean) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <C : MutableCollection<in Char>> CharArray.filterIndexedTo(destination: C, predicate: (index: Int, Char) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Returns a list containing all elements that are instances of specified type parameter R.\n * \n * @sample samples.collections.Collections.Filtering.filterIsInstance\n */\npublic inline fun <reified R> Array<*>.filterIsInstance(): List<@kotlin.internal.NoInfer R> {\n   return filterIsInstanceTo(ArrayList<R>())\n}\n\n/**\n * Appends all elements that are instances of specified type parameter R to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterIsInstanceTo\n */\npublic inline fun <reified R, C :

MutableCollection<in R>> Array<*>.filterIsInstanceTo(destination: C): C {\n   for (element in this) if (element is R) destination.add(element)\n   return destination\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun <T> Array<out T>.filterNot(predicate: (T) -> Boolean): List<T> {\n   return filterNotTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun ByteArray.filterNot(predicate: (Byte) -> Boolean): List<Byte> {\n   return filterNotTo(ArrayList<Byte>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun ShortArray.filterNot(predicate: (Short) -> Boolean): List<Short> {\n   return filterNotTo(ArrayList<Short>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun IntArray.filterNot(predicate: (Int) -> Boolean): List<Int> {\n   return filterNotTo(ArrayList<Int>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun LongArray.filterNot(predicate: (Long) -> Boolean): List<Long> {\n   return filterNotTo(ArrayList<Long>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun FloatArray.filterNot(predicate: (Float) -> Boolean): List<Float> {\n   return filterNotTo(ArrayList<Float>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun DoubleArray.filterNot(predicate: (Double) -> Boolean): List<Double> {\n   return filterNotTo(ArrayList<Double>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun BooleanArray.filterNot(predicate: (Boolean) -> Boolean): List<Boolean> {\n   return filterNotTo(ArrayList<Boolean>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun CharArray.filterNot(predicate: (Char) -> Boolean): List<Char> {\n   return filterNotTo(ArrayList<Char>(), predicate)\n}\n\n/**\n * Returns a list containing all elements that are not `null`.\n * \n * @sample samples.collections.Collections.Filtering.filterNotNull\n */\npublic fun <T : Any> Array<out T?>.filterNotNull(): List<T> {\n   return filterNotNullTo(ArrayList<T>())\n}\n\n/**\n * Appends all elements that are not `null` to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterNotNullTo\n */\npublic fun <C : MutableCollection<in T>, T : Any> Array<out T?>.filterNotNullTo(destination: C): C {\n   for (element in this) if (element != null) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <T, C : MutableCollection<in T>> Array<out T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Byte>> ByteArray.filterNotTo(destination: C, predicate: (Byte) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Short>> ShortArray.filterNotTo(destination: C, predicate: (Short) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Int>> IntArray.filterNotTo(destination: C, predicate: (Int) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not

matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Long>> LongArray.filterNotTo(destination: C, predicate: (Long) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Float>> FloatArray.filterNotTo(destination: C, predicate: (Float) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Double>> DoubleArray.filterNotTo(destination: C, predicate: (Double) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Boolean>> BooleanArray.filterNotTo(destination: C, predicate: (Boolean) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Char>> CharArray.filterNotTo(destination: C, predicate: (Char) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <T, C : MutableCollection<in T>> Array<out T>.filterTo(destination: C, predicate: (T) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Byte>> ByteArray.filterTo(destination: C, predicate: (Byte) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Short>> ShortArray.filterTo(destination: C, predicate: (Short) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Int>> IntArray.filterTo(destination: C, predicate: (Int) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Long>> LongArray.filterTo(destination: C, predicate: (Long) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Float>> FloatArray.filterTo(destination: C, predicate: (Float) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Double>> DoubleArray.filterTo(destination: C, predicate: (Double) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Boolean>> BooleanArray.filterTo(destination: C, predicate: (Boolean) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching

the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Char>> CharArray.filterTo(destination: C, predicate: (Char) -> Boolean): C {\n for (element in this) if (predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic fun <T> Array<out T>.slice(indices: IntRange): List<T> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic fun ByteArray.slice(indices: IntRange): List<Byte> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic fun ShortArray.slice(indices: IntRange): List<Short> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic fun IntArray.slice(indices: IntRange): List<Int> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic fun LongArray.slice(indices: IntRange): List<Long> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic fun FloatArray.slice(indices: IntRange): List<Float> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic fun DoubleArray.slice(indices: IntRange): List<Double> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic fun BooleanArray.slice(indices: IntRange): List<Boolean> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic fun CharArray.slice(indices: IntRange): List<Char> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun <T> Array<out T>.slice(indices: Iterable<Int>): List<T> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<T>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun ByteArray.slice(indices: Iterable<Int>): List<Byte> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<Byte>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun ShortArray.slice(indices: Iterable<Int>): List<Short> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<Short>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun IntArray.slice(indices: Iterable<Int>): List<Int> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<Int>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun LongArray.slice(indices: Iterable<Int>): List<Long> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<Long>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun FloatArray.slice(indices: Iterable<Int>): List<Float> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<Float>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun DoubleArray.slice(indices: Iterable<Int>): List<Double> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<Double>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun BooleanArray.slice(indices: Iterable<Int>):

List<Boolean> {\n   val size = indices.collectionSizeOrDefault(10)\n   if (size == 0) return emptyList()\n   val list = ArrayList<Boolean>(size)\n   for (index in indices) {\n   list.add(get(index))\n   }\n   return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun CharArray.slice(indices: Iterable<Int>): List<Char> {\n   val size = indices.collectionSizeOrDefault(10)\n   if (size == 0) return emptyList()\n   val list = ArrayList<Char>(size)\n   for (index in indices) {\n   list.add(get(index))\n   }\n   return list\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun <T> Array<T>.sliceArray(indices: Collection<Int>): Array<T> {\n   val result = arrayOfNulls(this, indices.size)\n   var targetIndex = 0\n   for (sourceIndex in indices) {\n   result[targetIndex++] = this[sourceIndex]\n   }\n   return result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun ByteArray.sliceArray(indices: Collection<Int>): ByteArray {\n   val result = ByteArray(indices.size)\n   var targetIndex = 0\n   for (sourceIndex in indices) {\n   result[targetIndex++] = this[sourceIndex]\n   }\n   return result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun ShortArray.sliceArray(indices: Collection<Int>): ShortArray {\n   val result = ShortArray(indices.size)\n   var targetIndex = 0\n   for (sourceIndex in indices) {\n   result[targetIndex++] = this[sourceIndex]\n   }\n   return result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun IntArray.sliceArray(indices: Collection<Int>): IntArray {\n   val result = IntArray(indices.size)\n   var targetIndex = 0\n   for (sourceIndex in indices) {\n   result[targetIndex++] = this[sourceIndex]\n   }\n   return result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun LongArray.sliceArray(indices: Collection<Int>): LongArray {\n   val result = LongArray(indices.size)\n   var targetIndex = 0\n   for (sourceIndex in indices) {\n   result[targetIndex++] = this[sourceIndex]\n   }\n   return result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun FloatArray.sliceArray(indices: Collection<Int>): FloatArray {\n   val result = FloatArray(indices.size)\n   var targetIndex = 0\n   for (sourceIndex in indices) {\n   result[targetIndex++] = this[sourceIndex]\n   }\n   return result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun DoubleArray.sliceArray(indices: Collection<Int>): DoubleArray {\n   val result = DoubleArray(indices.size)\n   var targetIndex = 0\n   for (sourceIndex in indices) {\n   result[targetIndex++] = this[sourceIndex]\n   }\n   return result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun BooleanArray.sliceArray(indices: Collection<Int>): BooleanArray {\n   val result = BooleanArray(indices.size)\n   var targetIndex = 0\n   for (sourceIndex in indices) {\n   result[targetIndex++] = this[sourceIndex]\n   }\n   return result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun CharArray.sliceArray(indices: Collection<Int>): CharArray {\n   val result = CharArray(indices.size)\n   var targetIndex = 0\n   for (sourceIndex in indices) {\n   result[targetIndex++] = this[sourceIndex]\n   }\n   return result\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\npublic fun <T> Array<T>.sliceArray(indices: IntRange): Array<T> {\n   if (indices.isEmpty()) return copyOfRange(0, 0)\n   return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\npublic fun ByteArray.sliceArray(indices: IntRange): ByteArray {\n   if (indices.isEmpty()) return ByteArray(0)\n   return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\npublic fun ShortArray.sliceArray(indices: IntRange): ShortArray {\n   if (indices.isEmpty()) return ShortArray(0)\n   return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\npublic fun IntArray.sliceArray(indices: IntRange): IntArray {\n   if (indices.isEmpty()) return IntArray(0)\n   return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\npublic fun LongArray.sliceArray(indices: IntRange): LongArray {\n   if (indices.isEmpty()) return LongArray(0)\n   return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\npublic fun FloatArray.sliceArray(indices: IntRange): FloatArray {\n   if (indices.isEmpty()) return FloatArray(0)\n   return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\npublic fun

```
DoubleArray.sliceArray(indices: IntRange): DoubleArray {\n    if (indices.isEmpty()) return DoubleArray(0)\n    return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\npublic fun BooleanArray.sliceArray(indices: IntRange): BooleanArray {\n    if (indices.isEmpty()) return BooleanArray(0)\n    return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\npublic fun CharArray.sliceArray(indices: IntRange): CharArray {\n    if (indices.isEmpty()) return CharArray(0)\n    return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun <T> Array<out T>.take(n: Int): List<T> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return emptyList()\n    if (n >= size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<T>(n)\n    for (item in this) {\n        list.add(item)\n        if (++count == n)\n            break\n    }\n    return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun ByteArray.take(n: Int): List<Byte> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return emptyList()\n    if (n >= size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Byte>(n)\n    for (item in this) {\n        list.add(item)\n        if (++count == n)\n            break\n    }\n    return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun ShortArray.take(n: Int): List<Short> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return emptyList()\n    if (n >= size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Short>(n)\n    for (item in this) {\n        list.add(item)\n        if (++count == n)\n            break\n    }\n    return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun IntArray.take(n: Int): List<Int> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return emptyList()\n    if (n >= size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Int>(n)\n    for (item in this) {\n        list.add(item)\n        if (++count == n)\n            break\n    }\n    return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun LongArray.take(n: Int): List<Long> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return emptyList()\n    if (n >= size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Long>(n)\n    for (item in this) {\n        list.add(item)\n        if (++count == n)\n            break\n    }\n    return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun FloatArray.take(n: Int): List<Float> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return emptyList()\n    if (n >= size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Float>(n)\n    for (item in this) {\n        list.add(item)\n        if (++count == n)\n            break\n    }\n    return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun DoubleArray.take(n: Int): List<Double> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return emptyList()\n    if (n >= size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Double>(n)\n    for (item in this) {\n        list.add(item)\n        if (++count == n)\n            break\n    }\n    return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun BooleanArray.take(n: Int): List<Boolean> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return emptyList()\n    if (n >= size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Boolean>(n)\n    for (item in this) {\n        list.add(item)\n        if (++count == n)\n            break\n    }\n    return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if
```

[n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun CharArray.take(n: Int): List<Char> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n if (n >= size) return toList()\n if (n == 1) return listOf(this[0])\n var count = 0\n val list = ArrayList<Char>(n)\n for (item in this) {\n list.add(item)\n if (++count == n)\n break\n }\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun <T> Array<out T>.takeLast(n: Int): List<T> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<T>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun ByteArray.takeLast(n: Int): List<Byte> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Byte>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun ShortArray.takeLast(n: Int): List<Short> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Short>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun IntArray.takeLast(n: Int): List<Int> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Int>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun LongArray.takeLast(n: Int): List<Long> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Long>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun FloatArray.takeLast(n: Int): List<Float> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Float>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun DoubleArray.takeLast(n: Int): List<Double> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Double>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun BooleanArray.takeLast(n: Int): List<Boolean> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Boolean>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun CharArray.takeLast(n: Int): List<Char> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return

```
emptyList()\n    val size = size\n    if (n >= size) return toList()\n    if (n == 1) return listOf(this[size - 1])\n    val list = ArrayList<Char>(n)\n    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun <T> Array<out T>.takeLastWhile(predicate: (T) -> Boolean): List<T> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun ByteArray.takeLastWhile(predicate: (Byte) -> Boolean): List<Byte> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun ShortArray.takeLastWhile(predicate: (Short) -> Boolean): List<Short> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun IntArray.takeLastWhile(predicate: (Int) -> Boolean): List<Int> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun LongArray.takeLastWhile(predicate: (Long) -> Boolean): List<Long> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun FloatArray.takeLastWhile(predicate: (Float) -> Boolean): List<Float> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun DoubleArray.takeLastWhile(predicate: (Double) -> Boolean): List<Double> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun BooleanArray.takeLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun CharArray.takeLastWhile(predicate: (Char) -> Boolean): List<Char> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun <T> Array<out T>.takeWhile(predicate: (T) -> Boolean): List<T> {\n    val list = ArrayList<T>()\n    for (item in this) {\n        if (!predicate(item))\n            break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun ByteArray.takeWhile(predicate: (Byte) -> Boolean): List<Byte> {\n    val list = ArrayList<Byte>()\n    for (item in this) {\n        if (!predicate(item))\n            break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun ShortArray.takeWhile(predicate: (Short) -> Boolean): List<Short> {\n    val list = ArrayList<Short>()\n    for (item in this) {\n        if (!predicate(item))\n            break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun IntArray.takeWhile(predicate: (Int) -> Boolean): List<Int> {\n    val list = ArrayList<Int>()\n    for (item in this) {\n        if (!predicate(item))\n
```

break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun LongArray.takeWhile(predicate: (Long) -> Boolean): List<Long> {\n val list = ArrayList<Long>()\n for (item in this) {\n if (!predicate(item))\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun FloatArray.takeWhile(predicate: (Float) -> Boolean): List<Float> {\n val list = ArrayList<Float>()\n for (item in this) {\n if (!predicate(item))\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun DoubleArray.takeWhile(predicate: (Double) -> Boolean): List<Double> {\n val list = ArrayList<Double>()\n for (item in this) {\n if (!predicate(item))\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun BooleanArray.takeWhile(predicate: (Boolean) -> Boolean): List<Boolean> {\n val list = ArrayList<Boolean>()\n for (item in this) {\n if (!predicate(item))\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun CharArray.takeWhile(predicate: (Char) -> Boolean): List<Char> {\n val list = ArrayList<Char>()\n for (item in this) {\n if (!predicate(item))\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Reverses elements in the array in-place.\n */\npublic fun <T> Array<T>.reverse(): Unit {\n val midPoint = (size / 2) - 1\n if (midPoint < 0) return\n var reverseIndex = lastIndex\n for (index in 0..midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\npublic fun ByteArray.reverse(): Unit {\n val midPoint = (size / 2) - 1\n if (midPoint < 0) return\n var reverseIndex = lastIndex\n for (index in 0..midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\npublic fun ShortArray.reverse(): Unit {\n val midPoint = (size / 2) - 1\n if (midPoint < 0) return\n var reverseIndex = lastIndex\n for (index in 0..midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\npublic fun IntArray.reverse(): Unit {\n val midPoint = (size / 2) - 1\n if (midPoint < 0) return\n var reverseIndex = lastIndex\n for (index in 0..midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\npublic fun LongArray.reverse(): Unit {\n val midPoint = (size / 2) - 1\n if (midPoint < 0) return\n var reverseIndex = lastIndex\n for (index in 0..midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\npublic fun FloatArray.reverse(): Unit {\n val midPoint = (size / 2) - 1\n if (midPoint < 0) return\n var reverseIndex = lastIndex\n for (index in 0..midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\npublic fun DoubleArray.reverse(): Unit {\n val midPoint = (size / 2) - 1\n if (midPoint < 0) return\n var reverseIndex = lastIndex\n for (index in 0..midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\npublic fun BooleanArray.reverse(): Unit {\n val midPoint = (size / 2) - 1\n if (midPoint < 0) return\n var reverseIndex = lastIndex\n for (index in 0..midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\npublic fun CharArray.reverse(): Unit {\n val midPoint = (size / 2) - 1\n if (midPoint < 0) return\n var reverseIndex = lastIndex\n for (index in 0..midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if

[fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Array<T>.reverse(fromIndex: Int, toIndex: Int): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   val midPoint = (fromIndex + toIndex) / 2\n   if (fromIndex == midPoint) return\n   var reverseIndex = toIndex - 1\n   for (index in fromIndex until midPoint) {\n   val tmp = this[index]\n   this[index] = this[reverseIndex]\n   this[reverseIndex] = tmp\n   reverseIndex--\n   }\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun ByteArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   val midPoint = (fromIndex + toIndex) / 2\n   if (fromIndex == midPoint) return\n   var reverseIndex = toIndex - 1\n   for (index in fromIndex until midPoint) {\n   val tmp = this[index]\n   this[index] = this[reverseIndex]\n   this[reverseIndex] = tmp\n   reverseIndex--\n   }\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun ShortArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   val midPoint = (fromIndex + toIndex) / 2\n   if (fromIndex == midPoint) return\n   var reverseIndex = toIndex - 1\n   for (index in fromIndex until midPoint) {\n   val tmp = this[index]\n   this[index] = this[reverseIndex]\n   this[reverseIndex] = tmp\n   reverseIndex--\n   }\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun IntArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   val midPoint = (fromIndex + toIndex) / 2\n   if (fromIndex == midPoint) return\n   var reverseIndex = toIndex - 1\n   for (index in fromIndex until midPoint) {\n   val tmp = this[index]\n   this[index] = this[reverseIndex]\n   this[reverseIndex] = tmp\n   reverseIndex--\n   }\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun LongArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   val midPoint = (fromIndex + toIndex) / 2\n   if (fromIndex == midPoint) return\n   var reverseIndex = toIndex - 1\n   for (index in fromIndex until midPoint) {\n   val tmp = this[index]\n   this[index] = this[reverseIndex]\n   this[reverseIndex] = tmp\n   reverseIndex--\n   }\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun FloatArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   val midPoint = (fromIndex + toIndex) / 2\n   if (fromIndex == midPoint) return\n   var reverseIndex = toIndex - 1\n   for (index in fromIndex until midPoint) {\n   val tmp = this[index]\n   this[index] = this[reverseIndex]\n   this[reverseIndex] = tmp\n   reverseIndex--\n   }\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if

[fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun DoubleArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n val midPoint = (fromIndex + toIndex) / 2\n if (fromIndex == midPoint) return\n var reverseIndex = toIndex - 1\n for (index in fromIndex until midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun BooleanArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n val midPoint = (fromIndex + toIndex) / 2\n if (fromIndex == midPoint) return\n var reverseIndex = toIndex - 1\n for (index in fromIndex until midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun CharArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n val midPoint = (fromIndex + toIndex) / 2\n if (fromIndex == midPoint) return\n var reverseIndex = toIndex - 1\n for (index in fromIndex until midPoint) {\n val tmp = this[index]\n this[index] = this[reverseIndex]\n this[reverseIndex] = tmp\n reverseIndex--\n }\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun <T> Array<out T>.reversed(): List<T> {\n if (isEmpty()) return emptyList()\n val list = toMutableList()\n list.reverse()\n return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun ByteArray.reversed(): List<Byte> {\n if (isEmpty()) return emptyList()\n val list = toMutableList()\n list.reverse()\n return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun ShortArray.reversed(): List<Short> {\n if (isEmpty()) return emptyList()\n val list = toMutableList()\n list.reverse()\n return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun IntArray.reversed(): List<Int> {\n if (isEmpty()) return emptyList()\n val list = toMutableList()\n list.reverse()\n return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun LongArray.reversed(): List<Long> {\n if (isEmpty()) return emptyList()\n val list = toMutableList()\n list.reverse()\n return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun FloatArray.reversed(): List<Float> {\n if (isEmpty()) return emptyList()\n val list = toMutableList()\n list.reverse()\n return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun DoubleArray.reversed(): List<Double> {\n if (isEmpty()) return emptyList()\n val list = toMutableList()\n list.reverse()\n return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun BooleanArray.reversed(): List<Boolean> {\n if (isEmpty()) return emptyList()\n val list = toMutableList()\n list.reverse()\n return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun CharArray.reversed(): List<Char> {\n if (isEmpty()) return emptyList()\n val list = toMutableList()\n list.reverse()\n return list\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun <T> Array<T>.reversedArray(): Array<T> {\n if (isEmpty()) return this\n val result = arrayOfNulls(this, size)\n val lastIndex = lastIndex\n for (i in 0..lastIndex)\n result[lastIndex - i] = this[i]\n return result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun ByteArray.reversedArray(): ByteArray {\n if (isEmpty()) return this\n val result = ByteArray(size)\n val lastIndex = lastIndex\n for (i in 0..lastIndex)\n result[lastIndex - i] = this[i]\n return result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun ShortArray.reversedArray(): ShortArray {\n if (isEmpty()) return this\n val result = ShortArray(size)\n val lastIndex = lastIndex\n for (i in 0..lastIndex)\n result[lastIndex - i] = this[i]\n return result\n}\n\n/**\n * Returns an array with elements of this

array in reversed order.\n */\npublic fun IntArray.reversedArray(): IntArray {\n if (isEmpty()) return this\n val result = IntArray(size)\n val lastIndex = lastIndex\n for (i in 0..lastIndex)\n result[lastIndex - i] = this[i]\n return result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun LongArray.reversedArray(): LongArray {\n if (isEmpty()) return this\n val result = LongArray(size)\n val lastIndex = lastIndex\n for (i in 0..lastIndex)\n result[lastIndex - i] = this[i]\n return result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun FloatArray.reversedArray(): FloatArray {\n if (isEmpty()) return this\n val result = FloatArray(size)\n val lastIndex = lastIndex\n for (i in 0..lastIndex)\n result[lastIndex - i] = this[i]\n return result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun DoubleArray.reversedArray(): DoubleArray {\n if (isEmpty()) return this\n val result = DoubleArray(size)\n val lastIndex = lastIndex\n for (i in 0..lastIndex)\n result[lastIndex - i] = this[i]\n return result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun BooleanArray.reversedArray(): BooleanArray {\n if (isEmpty()) return this\n val result = BooleanArray(size)\n val lastIndex = lastIndex\n for (i in 0..lastIndex)\n result[lastIndex - i] = this[i]\n return result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun CharArray.reversedArray(): CharArray {\n if (isEmpty()) return this\n val result = CharArray(size)\n val lastIndex = lastIndex\n for (i in 0..lastIndex)\n result[lastIndex - i] = this[i]\n return result\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Array<T>.shuffle(): Unit {\n shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\npublic fun ByteArray.shuffle(): Unit {\n shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\npublic fun ShortArray.shuffle(): Unit {\n shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\npublic fun IntArray.shuffle(): Unit {\n shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\npublic fun LongArray.shuffle(): Unit {\n shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\npublic fun FloatArray.shuffle(): Unit {\n shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\npublic fun DoubleArray.shuffle(): Unit {\n shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\npublic fun BooleanArray.shuffle(): Unit {\n shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\npublic fun CharArray.shuffle(): Unit {\n shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Array<T>.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1) {\n val j = random.nextInt(i + 1)\n val copy = this[i]\n this[i] = this[j]\n this[j] = copy\n }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\npublic fun ByteArray.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1) {\n val j = random.nextInt(i + 1)\n val copy = this[i]\n this[i] = this[j]\n this[j] = copy\n }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\npublic fun ShortArray.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1) {\n val j = random.nextInt(i + 1)\n val copy = this[i]\n this[i] = this[j]\n this[j] = copy\n }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\npublic fun IntArray.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1)

{\n val j = random.nextInt(i + 1)\n val copy = this[i]\n this[i] = this[j]\n this[j] = copy\n }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\npublic fun LongArray.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1) {\n val j = random.nextInt(i + 1)\n val copy = this[i]\n this[i] = this[j]\n this[j] = copy\n }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\npublic fun FloatArray.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1) {\n val j = random.nextInt(i + 1)\n val copy = this[i]\n this[i] = this[j]\n this[j] = copy\n }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\npublic fun DoubleArray.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1) {\n val j = random.nextInt(i + 1)\n val copy = this[i]\n this[i] = this[j]\n this[j] = copy\n }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\npublic fun BooleanArray.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1) {\n val j = random.nextInt(i + 1)\n val copy = this[i]\n this[i] = this[j]\n this[j] = copy\n }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\npublic fun CharArray.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1) {\n val j = random.nextInt(i + 1)\n val copy = this[i]\n this[i] = this[j]\n this[j] = copy\n }\n}\n\n/**\n * Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic inline fun <T, R : Comparable<R>> Array<out T>.sortBy(crossinline selector: (T) -> R?): Unit {\n if (size > 1) sortWith(compareBy(selector))\n}\n\n/**\n * Sorts elements in the array in-place descending according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic inline fun <T, R : Comparable<R>> Array<out T>.sortByDescending(crossinline selector: (T) -> R?): Unit {\n if (size > 1) sortWith(compareByDescending(selector))\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T : Comparable<T>> Array<out T>.sortDescending(): Unit {\n sortWith(reverseOrder())\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\npublic fun ByteArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\npublic fun ShortArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\npublic fun IntArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\npublic fun LongArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\npublic fun FloatArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\npublic fun DoubleArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n

*/\npublic fun CharArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T : Comparable<T>> Array<out T>.sorted(): List<T> {\n return sortedArray().asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\npublic fun ByteArray.sorted(): List<Byte> {\n return toTypedArray().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\npublic fun ShortArray.sorted(): List<Short> {\n return toTypedArray().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\npublic fun IntArray.sorted(): List<Int> {\n return toTypedArray().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\npublic fun LongArray.sorted(): List<Long> {\n return toTypedArray().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\npublic fun FloatArray.sorted(): List<Float> {\n return toTypedArray().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\npublic fun DoubleArray.sorted(): List<Double> {\n return toTypedArray().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\npublic fun CharArray.sorted(): List<Char> {\n return toTypedArray().apply { sort() }.asList()\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T : Comparable<T>> Array<T>.sortedArray(): Array<T> {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\npublic fun ByteArray.sortedArray(): ByteArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\npublic fun ShortArray.sortedArray(): ShortArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\npublic fun IntArray.sortedArray(): IntArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\npublic fun LongArray.sortedArray(): LongArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\npublic fun FloatArray.sortedArray(): FloatArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\npublic fun DoubleArray.sortedArray(): DoubleArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\npublic fun CharArray.sortedArray(): CharArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T : Comparable<T>> Array<T>.sortedArrayDescending(): Array<T> {\n if (isEmpty()) return this\n return this.copyOf().apply { sortWith(reverseOrder()) }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\npublic fun ByteArray.sortedArrayDescending(): ByteArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\npublic fun ShortArray.sortedArrayDescending(): ShortArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\npublic fun IntArray.sortedArrayDescending(): IntArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\npublic fun LongArray.sortedArrayDescending(): LongArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\npublic fun FloatArray.sortedArrayDescending(): FloatArray {\n if (isEmpty()) return this\n return this.copyOf().apply {

sortDescending() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\npublic fun DoubleArray.sortedArrayDescending(): DoubleArray {\n   if (isEmpty()) return this\n   return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\npublic fun CharArray.sortedArrayDescending(): CharArray {\n   if (isEmpty()) return this\n   return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according the specified [comparator].\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T> Array<out T>.sortedArrayWith(comparator: Comparator<in T>): Array<out T> {\n   if (isEmpty()) return this\n   return this.copyOf().apply { sortWith(comparator) }\n}\n\n/**\n * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <T, R : Comparable<R>> Array<out T>.sortedBy(crossinline selector: (T) -> R?): List<T> {\n   return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <R : Comparable<R>> ByteArray.sortedBy(crossinline selector: (Byte) -> R?): List<Byte> {\n   return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <R : Comparable<R>> ShortArray.sortedBy(crossinline selector: (Short) -> R?): List<Short> {\n   return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <R : Comparable<R>> IntArray.sortedBy(crossinline selector: (Int) -> R?): List<Int> {\n   return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <R : Comparable<R>> LongArray.sortedBy(crossinline selector: (Long) -> R?): List<Long> {\n   return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <R : Comparable<R>> FloatArray.sortedBy(crossinline selector: (Float) -> R?): List<Float> {\n   return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <R : Comparable<R>> DoubleArray.sortedBy(crossinline selector: (Double) -> R?): List<Double> {\n   return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <R : Comparable<R>> BooleanArray.sortedBy(crossinline selector: (Boolean) -> R?): List<Boolean> {\n   return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <R : Comparable<R>> CharArray.sortedBy(crossinline selector: (Char) -> R?): List<Char> {\n   return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic inline fun <T, R : Comparable<R>>

Array<out T>.sortedByDescending(crossinline selector: (T) -> R?): List<T> {\n   return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.\n */\npublic inline fun <R : Comparable<R>> ByteArray.sortedByDescending(crossinline selector: (Byte) -> R?): List<Byte> {\n   return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.\n */\npublic inline fun <R : Comparable<R>> ShortArray.sortedByDescending(crossinline selector: (Short) -> R?): List<Short> {\n   return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.\n */\npublic inline fun <R : Comparable<R>> IntArray.sortedByDescending(crossinline selector: (Int) -> R?): List<Int> {\n   return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.\n */\npublic inline fun <R : Comparable<R>> LongArray.sortedByDescending(crossinline selector: (Long) -> R?): List<Long> {\n   return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.\n */\npublic inline fun <R : Comparable<R>> FloatArray.sortedByDescending(crossinline selector: (Float) -> R?): List<Float> {\n   return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.\n */\npublic inline fun <R : Comparable<R>> DoubleArray.sortedByDescending(crossinline selector: (Double) -> R?): List<Double> {\n   return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.\n */\npublic inline fun <R : Comparable<R>> BooleanArray.sortedByDescending(crossinline selector: (Boolean) -> R?): List<Boolean> {\n   return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.\n */\npublic inline fun <R : Comparable<R>> CharArray.sortedByDescending(crossinline selector: (Char) -> R?): List<Char> {\n   return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T : Comparable<T>> Array<out T>.sortedDescending(): List<T> {\n   return sortedWith(reverseOrder())\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n */\npublic fun ByteArray.sortedDescending(): List<Byte> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n */\npublic fun ShortArray.sortedDescending(): List<Short> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n */\npublic fun IntArray.sortedDescending(): List<Int> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n */\npublic fun LongArray.sortedDescending(): List<Long> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n */\npublic fun FloatArray.sortedDescending(): List<Float> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n */\npublic fun DoubleArray.sortedDescending(): List<Double> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n */\npublic fun CharArray.sortedDescending(): List<Char> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted according to the specified [comparator].\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T> Array<out T>.sortedWith(comparator: Comparator<in T>): List<T> {\n   return sortedArrayWith(comparator).asList()\n}\n\n/**\n * Returns a list of all elements sorted according to the specified [comparator].\n */\npublic fun ByteArray.sortedWith(comparator: Comparator<in Byte>): List<Byte> {\n   return toTypedArray().apply { sortWith(comparator) }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according

to the specified [comparator].\n */\npublic fun ShortArray.sortedWith(comparator: Comparator<in Short>): List<Short> {\n   return toTypedArray().apply { sortWith(comparator) }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to the specified [comparator].\n */\npublic fun IntArray.sortedWith(comparator: Comparator<in Int>): List<Int> {\n   return toTypedArray().apply { sortWith(comparator) }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to the specified [comparator].\n */\npublic fun LongArray.sortedWith(comparator: Comparator<in Long>): List<Long> {\n   return toTypedArray().apply { sortWith(comparator) }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to the specified [comparator].\n */\npublic fun FloatArray.sortedWith(comparator: Comparator<in Float>): List<Float> {\n   return toTypedArray().apply { sortWith(comparator) }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to the specified [comparator].\n */\npublic fun DoubleArray.sortedWith(comparator: Comparator<in Double>): List<Double> {\n   return toTypedArray().apply { sortWith(comparator) }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to the specified [comparator].\n */\npublic fun BooleanArray.sortedWith(comparator: Comparator<in Boolean>): List<Boolean> {\n   return toTypedArray().apply { sortWith(comparator) }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to the specified [comparator].\n */\npublic fun CharArray.sortedWith(comparator: Comparator<in Char>): List<Char> {\n   return toTypedArray().apply { sortWith(comparator) }.asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun <T> Array<out T>.asList(): List<T>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun ByteArray.asList(): List<Byte>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun ShortArray.asList(): List<Short>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun IntArray.asList(): List<Int>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun LongArray.asList(): List<Long>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun FloatArray.asList(): List<Float>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun DoubleArray.asList(): List<Double>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun BooleanArray.asList(): List<Boolean>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun CharArray.asList(): List<Char>\n\n/**\n * Returns `true` if the two specified arrays are *deeply* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * If two corresponding elements are nested arrays, they are also compared deeply.\n * If any of arrays contains itself on any nesting level the behavior is undefined.\n * \n * The elements of other types are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.LowPriorityInOverloadResolution\npublic expect infix fun <T> Array<out T>.contentDeepEquals(other: Array<out T>): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *deeply* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The specified arrays are also considered deeply equal if both are `null`.\n * \n * If two corresponding elements are nested arrays, they are also compared deeply.\n * If any of arrays contains itself on any nesting level the behavior is undefined.\n * \n * The elements of other types are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\npublic expect infix fun <T> Array<out T>?.contentDeepEquals(other: Array<out T>?): Boolean\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n * Nested arrays are treated as lists too.\n * \n * If any of arrays contains itself on any nesting level the behavior is undefined.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.LowPriorityInOverloadResolution\npublic expect fun <T> Array<out T>.contentDeepHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n * Nested arrays are treated as lists too.\n * \n * If any of arrays contains itself on any nesting level the behavior is undefined.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun <T> Array<out T>?.contentDeepHashCode(): Int\n\n/**\n * Returns a string representation of the contents of this array as if it is a [List].\n * Nested arrays are treated as lists too.\n * \n * If any of arrays contains itself on any nesting level that reference\n * is rendered as `\"[...]\"` to prevent recursion.\n * \n * @sample samples.collections.Arrays.ContentOperations.contentDeepToString\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.LowPriorityInOverloadResolution\npublic expect fun <T> Array<out

T>.contentDeepToString(): String\n\n/**\n * Returns a string representation of the contents of this array as if it is a [List].\n * Nested arrays are treated as lists too.\n * \n * If any of arrays contains itself on any nesting level that reference\n * is rendered as `\"[...]\"` to prevent recursion.\n * \n * @sample samples.collections.Arrays.ContentOperations.contentDeepToString\n */\n@SinceKotlin(\"1.4\")\npublic expect fun <T> Array<out T>?.contentDeepToString(): String\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect infix fun <T> Array<out T>.contentEquals(other: Array<out T>): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect infix fun ByteArray.contentEquals(other: ByteArray): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect infix fun ShortArray.contentEquals(other: ShortArray): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect infix fun IntArray.contentEquals(other: IntArray): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect infix fun LongArray.contentEquals(other: LongArray): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect infix fun FloatArray.contentEquals(other: FloatArray): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean\n\n/**\n * Returns `true` if the two specified arrays are

structurally equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect infix fun CharArray.contentEquals(other: CharArray): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\npublic expect infix fun <T> Array<out T>?.contentEquals(other: Array<out T>?): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\npublic expect infix fun ByteArray?.contentEquals(other: ByteArray?): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\npublic expect infix fun ShortArray?.contentEquals(other: ShortArray?): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\npublic expect infix fun IntArray?.contentEquals(other: IntArray?): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\npublic expect infix fun LongArray?.contentEquals(other: LongArray?): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\npublic expect infix fun FloatArray?.contentEquals(other: FloatArray?): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\npublic expect infix fun DoubleArray?.contentEquals(other: DoubleArray?): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\npublic expect infix fun BooleanArray?.contentEquals(other: BooleanArray?): Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\npublic expect infix fun CharArray?.contentEquals(other: CharArray?): Boolean\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun <T> Array<out T>.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun ByteArray.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is

[List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun ShortArray.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun IntArray.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun LongArray.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun FloatArray.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun DoubleArray.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun BooleanArray.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun CharArray.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\npublic expect fun <T> Array<out T>?.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\npublic expect fun ByteArray?.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\npublic expect fun ShortArray?.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\npublic expect fun IntArray?.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\npublic expect fun LongArray?.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\npublic expect fun FloatArray?.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\npublic expect fun DoubleArray?.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\npublic expect fun BooleanArray?.contentHashCode(): Int\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\npublic expect fun CharArray?.contentHashCode(): Int\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun <T> Array<out T>.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun ByteArray.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun ShortArray.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation

warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun IntArray.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun LongArray.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun FloatArray.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun DoubleArray.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun BooleanArray.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic expect fun CharArray.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\npublic expect fun <T> Array<out T>?.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\npublic expect fun ByteArray?.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\npublic expect fun ShortArray?.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\npublic expect fun IntArray?.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\npublic expect fun LongArray?.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\npublic expect fun FloatArray?.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\npublic expect fun DoubleArray?.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\npublic expect fun BooleanArray?.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\npublic expect fun CharArray?.contentToString(): String\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param

endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\npublic expect fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T>\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\npublic expect fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\npublic expect fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\npublic expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\npublic expect fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):

LongArray\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\npublic expect fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): FloatArray\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\npublic expect fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): DoubleArray\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\npublic expect fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): BooleanArray\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\npublic expect fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\n@Suppress(\"NO_ACTUAL_FOR_EXPECT\")\npublic expect fun <T> Array<T>.copyOf(): Array<T>\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic expect fun ByteArray.copyOf(): ByteArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample

samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic expect fun ShortArray.copyOf(): ShortArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic expect fun IntArray.copyOf(): IntArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic expect fun LongArray.copyOf(): LongArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic expect fun FloatArray.copyOf(): FloatArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic expect fun DoubleArray.copyOf(): DoubleArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic expect fun BooleanArray.copyOf(): BooleanArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic expect fun CharArray.copyOf(): CharArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun ByteArray.copyOf(newSize: Int): ByteArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun ShortArray.copyOf(newSize: Int): ShortArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun IntArray.copyOf(newSize: Int): IntArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun LongArray.copyOf(newSize: Int): LongArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun FloatArray.copyOf(newSize: Int): FloatArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun DoubleArray.copyOf(newSize: Int): DoubleArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with `false` values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `false` values.\n * \n * @sample

samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun BooleanArray.copyOf(newSize: Int): BooleanArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with null char (`\\u0000`) values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with null char (`\\u0000`) values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun CharArray.copyOf(newSize: Int): CharArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with `null` values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `null` values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizingCopyOf\n */\n@Suppress(\"NO_ACTUAL_FOR_EXPECT\")\npublic expect fun <T> Array<T>.copyOf(newSize: Int): Array<T?>\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@Suppress(\"NO_ACTUAL_FOR_EXPECT\")\npublic expect fun <T> Array<T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T>\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic expect fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic expect fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic expect fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic expect fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic expect fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic expect fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray\n\n/**\n * Returns a new array which is a

copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic expect fun BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic expect fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\npublic expect fun <T> Array<T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\npublic expect fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\npublic expect fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\npublic expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\npublic expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\npublic expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\npublic expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if

[fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\npublic expect fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\npublic expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Returns the range of valid indices for the array.\n */\npublic val <T> Array<out T>.indices: IntRange\n   get() = IntRange(0, lastIndex)\n\n/**\n * Returns the range of valid indices for the array.\n */\npublic val ByteArray.indices: IntRange\n   get() = IntRange(0, lastIndex)\n\n/**\n * Returns the range of valid indices for the array.\n */\npublic val ShortArray.indices: IntRange\n   get() = IntRange(0, lastIndex)\n\n/**\n * Returns the range of valid indices for the array.\n */\npublic val IntArray.indices: IntRange\n   get() = IntRange(0, lastIndex)\n\n/**\n * Returns the range of valid indices for the array.\n */\npublic val LongArray.indices: IntRange\n   get() = IntRange(0, lastIndex)\n\n/**\n * Returns the range of valid indices for the array.\n */\npublic val FloatArray.indices: IntRange\n   get() = IntRange(0, lastIndex)\n\n/**\n * Returns the range of valid indices for the array.\n */\npublic val DoubleArray.indices: IntRange\n   get() = IntRange(0, lastIndex)\n\n/**\n * Returns the range of valid indices for the array.\n */\npublic val BooleanArray.indices: IntRange\n   get() = IntRange(0, lastIndex)\n\n/**\n * Returns the range of valid indices for the array.\n */\npublic val CharArray.indices: IntRange\n   get() = IntRange(0, lastIndex)\n\n/**\n * Returns `true` if the array is empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.isEmpty(): Boolean {\n   return size == 0\n}\n\n/**\n * Returns `true` if the array is empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.isEmpty(): Boolean {\n   return size == 0\n}\n\n/**\n * Returns `true` if the array is empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.isEmpty(): Boolean {\n   return size == 0\n}\n\n/**\n * Returns `true` if the array is empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.isEmpty(): Boolean {\n   return size == 0\n}\n\n/**\n * Returns `true` if the array is empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.isEmpty(): Boolean {\n   return size == 0\n}\n\n/**\n * Returns `true` if the array is empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.isEmpty(): Boolean {\n   return size == 0\n}\n\n/**\n * Returns `true` if the array is empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.isEmpty(): Boolean {\n   return size == 0\n}\n\n/**\n * Returns `true` if the array is empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.isEmpty(): Boolean {\n   return size == 0\n}\n\n/**\n * Returns `true` if the array is empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.isEmpty(): Boolean {\n   return size == 0\n}\n\n/**\n * Returns `true` if the array is not empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.isNotEmpty(): Boolean {\n   return !isEmpty()\n}\n\n/**\n * Returns `true` if the array is not empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.isNotEmpty(): Boolean {\n   return !isEmpty()\n}\n\n/**\n * Returns `true` if the array is not empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.isNotEmpty(): Boolean {\n   return !isEmpty()\n}\n\n/**\n * Returns `true` if the array is not empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.isNotEmpty(): Boolean {\n   return !isEmpty()\n}\n\n/**\n * Returns `true` if the array is not empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.isNotEmpty(): Boolean {\n   return !isEmpty()\n}\n\n/**\n * Returns `true` if the array is not empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.isNotEmpty(): Boolean {\n   return !isEmpty()\n}\n\n/**\n * Returns `true` if the array is not empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.isNotEmpty(): Boolean {\n   return !isEmpty()\n}\n\n/**\n * Returns `true` if the array is not empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.isNotEmpty(): Boolean {\n   return !isEmpty()\n}\n\n/**\n * Returns `true` if the array is not empty.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.isNotEmpty(): Boolean {\n   return !isEmpty()\n}\n\n/**\n * Returns the last valid index for the array.\n */\npublic val <T> Array<out T>.lastIndex: Int\n   get() = size - 1\n\n/**\n * Returns the last valid index

for the array.\n */\npublic val ByteArray.lastIndex: Int\n   get() = size - 1\n\n/**\n * Returns the last valid index for the array.\n */\npublic val ShortArray.lastIndex: Int\n   get() = size - 1\n\n/**\n * Returns the last valid index for the array.\n */\npublic val IntArray.lastIndex: Int\n   get() = size - 1\n\n/**\n * Returns the last valid index for the array.\n */\npublic val LongArray.lastIndex: Int\n   get() = size - 1\n\n/**\n * Returns the last valid index for the array.\n */\npublic val FloatArray.lastIndex: Int\n   get() = size - 1\n\n/**\n * Returns the last valid index for the array.\n */\npublic val DoubleArray.lastIndex: Int\n   get() = size - 1\n\n/**\n * Returns the last valid index for the array.\n */\npublic val BooleanArray.lastIndex: Int\n   get() = size - 1\n\n/**\n * Returns the last valid index for the array.\n */\npublic val CharArray.lastIndex: Int\n   get() = size - 1\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"NO_ACTUAL_FOR_EXPECT\")\npublic expect operator fun <T> Array<T>.plus(element: T): Array<T>\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun ByteArray.plus(element: Byte): ByteArray\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun ShortArray.plus(element: Short): ShortArray\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun IntArray.plus(element: Int): IntArray\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun LongArray.plus(element: Long): LongArray\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun FloatArray.plus(element: Float): FloatArray\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun DoubleArray.plus(element: Double): DoubleArray\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun BooleanArray.plus(element: Boolean): BooleanArray\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun CharArray.plus(element: Char): CharArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n@Suppress(\"NO_ACTUAL_FOR_EXPECT\")\npublic expect operator fun <T> Array<T>.plus(elements: Collection<T>): Array<T>\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun ByteArray.plus(elements: Collection<Byte>): ByteArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun ShortArray.plus(elements: Collection<Short>): ShortArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun IntArray.plus(elements: Collection<Int>): IntArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun LongArray.plus(elements: Collection<Long>): LongArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun FloatArray.plus(elements: Collection<Float>): FloatArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun CharArray.plus(elements: Collection<Char>): CharArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@Suppress(\"NO_ACTUAL_FOR_EXPECT\")\npublic expect operator fun <T> Array<T>.plus(elements: Array<out T>): Array<T>\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun ByteArray.plus(elements: ByteArray): ByteArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun ShortArray.plus(elements: ShortArray):

ShortArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun IntArray.plus(elements: IntArray): IntArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun LongArray.plus(elements: LongArray): LongArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun FloatArray.plus(elements: FloatArray): FloatArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun CharArray.plus(elements: CharArray): CharArray\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"NO_ACTUAL_FOR_EXPECT\")\npublic expect fun <T> Array<T>.plusElement(element: T): Array<T>\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun IntArray.sort(): Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun LongArray.sort(): Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun ByteArray.sort(): Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun ShortArray.sort(): Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun DoubleArray.sort(): Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun FloatArray.sort(): Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun CharArray.sort(): Unit\n\n/**\n * Sorts the array in-place according to the natural order of its elements.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n * @sample samples.collections.Arrays.Sorting.sortArrayOfComparable\n */\npublic expect fun <T : Comparable<T>> Array<out T>.sort(): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArrayOfComparable\n */\n@SinceKotlin(\"1.4\")\npublic expect fun <T : Comparable<T>> Array<out T>.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\npublic expect fun ByteArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\npublic expect fun ShortArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *

@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\npublic expect fun IntArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\npublic expect fun LongArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\npublic expect fun FloatArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\npublic expect fun DoubleArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\npublic expect fun CharArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun <T : Comparable<T>> Array<out T>.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n   sortWith(reverseOrder(), fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun ByteArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n   sort(fromIndex, toIndex)\n   reverse(fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun ShortArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n   sort(fromIndex, toIndex)\n   reverse(fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n

*/\n@SinceKotlin(\"1.4\")\npublic fun IntArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n sort(fromIndex, toIndex)\n reverse(fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun LongArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n sort(fromIndex, toIndex)\n reverse(fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun FloatArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n sort(fromIndex, toIndex)\n reverse(fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun DoubleArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n sort(fromIndex, toIndex)\n reverse(fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\npublic fun CharArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n sort(fromIndex, toIndex)\n reverse(fromIndex, toIndex)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given [comparator].\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic expect fun <T> Array<out T>.sortWith(comparator: Comparator<in T>): Unit\n\n/**\n * Sorts a range in the array in-place with the given [comparator].\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic expect fun <T> Array<out T>.sortWith(comparator: Comparator<in T>, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Returns an array of Boolean containing all of the elements of this generic array.\n */\npublic fun Array<out Boolean>.toBooleanArray(): BooleanArray {\n return BooleanArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of Byte containing all of the elements of this generic array.\n */\npublic fun Array<out Byte>.toByteArray(): ByteArray {\n return ByteArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of Char containing all of the elements of this generic array.\n */\npublic fun Array<out Char>.toCharArray(): CharArray {\n return CharArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of Double containing all of the elements of this generic array.\n */\npublic fun Array<out Double>.toDoubleArray(): DoubleArray {\n return DoubleArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of Float containing all of the elements of this generic array.\n */\npublic fun Array<out Float>.toFloatArray(): FloatArray {\n return FloatArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of Int containing all of the elements of this generic array.\n */\npublic fun Array<out Int>.toIntArray(): IntArray {\n return IntArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of Long containing all of the elements of this generic array.\n */\npublic fun Array<out Long>.toLongArray(): LongArray {\n return LongArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of Short containing

all of the elements of this generic array.\n */\npublic fun Array<out Short>.toShortArray(): ShortArray {\n   return ShortArray(size) { index -> this[index] }\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic expect fun ByteArray.toTypedArray(): Array<Byte>\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic expect fun ShortArray.toTypedArray(): Array<Short>\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic expect fun IntArray.toTypedArray(): Array<Int>\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic expect fun LongArray.toTypedArray(): Array<Long>\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic expect fun FloatArray.toTypedArray(): Array<Float>\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic expect fun DoubleArray.toTypedArray(): Array<Double>\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic expect fun BooleanArray.toTypedArray(): Array<Boolean>\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic expect fun CharArray.toTypedArray(): Array<Char>\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives\n */\npublic inline fun <T, K, V> Array<out T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives\n */\npublic inline fun <K, V> ByteArray.associate(transform: (Byte) -> Pair<K, V>): Map<K, V> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives\n */\npublic inline fun <K, V> ShortArray.associate(transform: (Short) -> Pair<K, V>): Map<K, V> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives\n */\npublic inline fun <K, V> IntArray.associate(transform: (Int) -> Pair<K, V>): Map<K, V> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives\n */\npublic inline fun <K, V> LongArray.associate(transform: (Long) -> Pair<K, V>): Map<K, V> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives\n */\npublic inline fun <K, V> FloatArray.associate(transform: (Float) -> Pair<K, V>): Map<K, V> {\n   val capacity =

mapCapacity(size).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives\n */\npublic inline fun <K, V> DoubleArray.associate(transform: (Double) -> Pair<K, V>): Map<K, V> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives\n */\npublic inline fun <K, V> BooleanArray.associate(transform: (Boolean) -> Pair<K, V>): Map<K, V> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives\n */\npublic inline fun <K, V> CharArray.associate(transform: (Char) -> Pair<K, V>): Map<K, V> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy\n */\npublic inline fun <T, K> Array<out T>.associateBy(keySelector: (T) -> K): Map<K, T> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy\n */\npublic inline fun <K> ByteArray.associateBy(keySelector: (Byte) -> K): Map<K, Byte> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, Byte>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy\n */\npublic inline fun <K> ShortArray.associateBy(keySelector: (Short) -> K): Map<K, Short> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, Short>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy\n */\npublic inline fun <K> IntArray.associateBy(keySelector: (Int) -> K): Map<K, Int> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, Int>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n *

returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy\n */\npublic inline fun <K> LongArray.associateBy(keySelector: (Long) -> K): Map<K, Long> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, Long>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy\n */\npublic inline fun <K> FloatArray.associateBy(keySelector: (Float) -> K): Map<K, Float> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, Float>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy\n */\npublic inline fun <K> DoubleArray.associateBy(keySelector: (Double) -> K): Map<K, Double> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, Double>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy\n */\npublic inline fun <K> BooleanArray.associateBy(keySelector: (Boolean) -> K): Map<K, Boolean> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, Boolean>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy\n */\npublic inline fun <K> CharArray.associateBy(keySelector: (Char) -> K): Map<K, Char> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, Char>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */\npublic inline fun <T, K, V> Array<out T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */\npublic inline fun <K, V> ByteArray.associateBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, V> {\n   val capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, V>(capacity),

keySelector, valueTransform)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */\npublic inline fun <K, V> ShortArray.associateBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */\npublic inline fun <K, V> IntArray.associateBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */\npublic inline fun <K, V> LongArray.associateBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */\npublic inline fun <K, V> FloatArray.associateBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */\npublic inline fun <K, V> DoubleArray.associateBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */\npublic inline fun <K, V> BooleanArray.associateBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */\npublic inline fun <K, V> CharArray.associateBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {\n val

capacity = mapCapacity(size).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */\npublic inline fun <T, K, M : MutableMap<in K, in T>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K): M {\n   for (element in this) {\n   destination.put(keySelector(element), element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */\npublic inline fun <K, M : MutableMap<in K, in Byte>> ByteArray.associateByTo(destination: M, keySelector: (Byte) -> K): M {\n   for (element in this) {\n   destination.put(keySelector(element), element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */\npublic inline fun <K, M : MutableMap<in K, in Short>> ShortArray.associateByTo(destination: M, keySelector: (Short) -> K): M {\n   for (element in this) {\n   destination.put(keySelector(element), element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */\npublic inline fun <K, M : MutableMap<in K, in Int>> IntArray.associateByTo(destination: M, keySelector: (Int) -> K): M {\n   for (element in this) {\n   destination.put(keySelector(element), element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */\npublic inline fun <K, M : MutableMap<in K, in Long>> LongArray.associateByTo(destination: M, keySelector: (Long) -> K): M {\n   for (element in this) {\n   destination.put(keySelector(element), element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */\npublic inline fun <K, M : MutableMap<in K, in Float>> FloatArray.associateByTo(destination: M, keySelector: (Float) -> K): M {\n   for (element in this) {\n   destination.put(keySelector(element), element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */\npublic inline fun <K, M : MutableMap<in K, in Double>> DoubleArray.associateByTo(destination: M, keySelector: (Double) -> K): M {\n   for (element in this) {\n   destination.put(keySelector(element), element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */\npublic

inline fun <K, M : MutableMap<in K, in Boolean>> BooleanArray.associateByTo(destination: M, keySelector: (Boolean) -> K): M {\n   for (element in this) {\n   destination.put(keySelector(element), element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */\npublic inline fun <K, M : MutableMap<in K, in Char>> CharArray.associateByTo(destination: M, keySelector: (Char) -> K): M {\n   for (element in this) {\n   destination.put(keySelector(element), element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform\n */\npublic inline fun <T, K, V, M : MutableMap<in K, in V>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {\n   for (element in this) {\n   destination.put(keySelector(element), valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> ByteArray.associateByTo(destination: M, keySelector: (Byte) -> K, valueTransform: (Byte) -> V): M {\n   for (element in this) {\n   destination.put(keySelector(element), valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> ShortArray.associateByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M {\n   for (element in this) {\n   destination.put(keySelector(element), valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> IntArray.associateByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M {\n   for (element in this) {\n   destination.put(keySelector(element), valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> LongArray.associateByTo(destination: M, keySelector: (Long) -> K, valueTransform: (Long) -> V): M {\n   for (element in this) {\n   destination.put(keySelector(element), valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform\n */\npublic

inline fun <K, V, M : MutableMap<in K, in V>> FloatArray.associateByTo(destination: M, keySelector: (Float) -> K, valueTransform: (Float) -> V): M {\n   for (element in this) {\n   destination.put(keySelector(element), valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> DoubleArray.associateByTo(destination: M, keySelector: (Double) -> K, valueTransform: (Double) -> V): M {\n   for (element in this) {\n destination.put(keySelector(element), valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> BooleanArray.associateByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M {\n   for (element in this) {\n destination.put(keySelector(element), valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> CharArray.associateByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {\n   for (element in this) {\n destination.put(keySelector(element), valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */\npublic inline fun <T, K, V, M : MutableMap<in K, in V>> Array<out T>.associateTo(destination: M, transform: (T) -> Pair<K, V>): M {\n   for (element in this) {\n   destination += transform(element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> ByteArray.associateTo(destination: M, transform: (Byte) -> Pair<K, V>): M {\n   for (element in this) {\n   destination += transform(element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> ShortArray.associateTo(destination: M, transform: (Short) -> Pair<K, V>): M {\n   for (element in this) {\n   destination += transform(element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> IntArray.associateTo(destination: M, transform: (Int) -> Pair<K, V>): M {\n   for (element in this) {\n   destination += transform(element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the

map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> LongArray.associateTo(destination: M, transform: (Long) -> Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> FloatArray.associateTo(destination: M, transform: (Float) -> Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> DoubleArray.associateTo(destination: M, transform: (Double) -> Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> BooleanArray.associateTo(destination: M, transform: (Boolean) -> Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> CharArray.associateTo(destination: M, transform: (Char) -> Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n return destination\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <K, V> Array<out K>.associateWith(valueSelector: (K) -> V): Map<K, V> {\n val result = LinkedHashMap<K, V>(mapCapacity(size).coerceAtLeast(16))\n return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V> ByteArray.associateWith(valueSelector: (Byte) -> V): Map<Byte, V> {\n val result = LinkedHashMap<Byte, V>(mapCapacity(size).coerceAtLeast(16))\n return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V> ShortArray.associateWith(valueSelector: (Short) -> V): Map<Short, V> {\n val result = LinkedHashMap<Short, V>(mapCapacity(size).coerceAtLeast(16))\n return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample

samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V> IntArray.associateWith(valueSelector: (Int) -> V): Map<Int, V> {\n   val result = LinkedHashMap<Int, V>(mapCapacity(size).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V> LongArray.associateWith(valueSelector: (Long) -> V): Map<Long, V> {\n   val result = LinkedHashMap<Long, V>(mapCapacity(size).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V> FloatArray.associateWith(valueSelector: (Float) -> V): Map<Float, V> {\n   val result = LinkedHashMap<Float, V>(mapCapacity(size).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V> DoubleArray.associateWith(valueSelector: (Double) -> V): Map<Double, V> {\n   val result = LinkedHashMap<Double, V>(mapCapacity(size).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V> BooleanArray.associateWith(valueSelector: (Boolean) -> V): Map<Boolean, V> {\n   val result = LinkedHashMap<Boolean, V>(mapCapacity(size).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V> CharArray.associateWith(valueSelector: (Char) -> V): Map<Char, V> {\n   val result = LinkedHashMap<Char, V>(mapCapacity(size.coerceAtMost(128)).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <K, V, M : MutableMap<in K, in V>> Array<out K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {\n   for (element in this) {\n   destination.put(element, valueSelector(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n *

@sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in Byte, in V>> ByteArray.associateWithTo(destination: M, valueSelector: (Byte) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in Short, in V>> ShortArray.associateWithTo(destination: M, valueSelector: (Short) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in Int, in V>> IntArray.associateWithTo(destination: M, valueSelector: (Int) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in Long, in V>> LongArray.associateWithTo(destination: M, valueSelector: (Long) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in Float, in V>> FloatArray.associateWithTo(destination: M, valueSelector: (Float) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in Double, in V>> DoubleArray.associateWithTo(destination: M, valueSelector: (Double) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in Boolean, in V>> BooleanArray.associateWithTo(destination: M, valueSelector: (Boolean) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements

are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in Char, in V>> CharArray.associateWithTo(destination: M, valueSelector: (Char) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n */\npublic fun <T, C : MutableCollection<in T>> Array<out T>.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n */\npublic fun <C : MutableCollection<in Byte>> ByteArray.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n */\npublic fun <C : MutableCollection<in Short>> ShortArray.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n */\npublic fun <C : MutableCollection<in Int>> IntArray.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n */\npublic fun <C : MutableCollection<in Long>> LongArray.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n */\npublic fun <C : MutableCollection<in Float>> FloatArray.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n */\npublic fun <C : MutableCollection<in Double>> DoubleArray.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n */\npublic fun <C : MutableCollection<in Boolean>> BooleanArray.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n */\npublic fun <C : MutableCollection<in Char>> CharArray.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n */\npublic fun <T> Array<out T>.toHashSet(): HashSet<T> {\n return toCollection(HashSet<T>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n */\npublic fun ByteArray.toHashSet(): HashSet<Byte> {\n return toCollection(HashSet<Byte>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n */\npublic fun ShortArray.toHashSet(): HashSet<Short> {\n return toCollection(HashSet<Short>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n */\npublic fun IntArray.toHashSet(): HashSet<Int> {\n return toCollection(HashSet<Int>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n */\npublic fun LongArray.toHashSet(): HashSet<Long> {\n return toCollection(HashSet<Long>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n */\npublic fun FloatArray.toHashSet(): HashSet<Float> {\n return toCollection(HashSet<Float>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n */\npublic fun DoubleArray.toHashSet(): HashSet<Double> {\n return toCollection(HashSet<Double>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n */\npublic fun BooleanArray.toHashSet(): HashSet<Boolean> {\n return toCollection(HashSet<Boolean>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n */\npublic fun CharArray.toHashSet(): HashSet<Char> {\n return toCollection(HashSet<Char>(mapCapacity(size.coerceAtMost(128))))\n}\n\n/**\n * Returns a [List] containing all elements.\n */\npublic fun <T> Array<out T>.toList(): List<T> {\n return when (size) {\n 0 -> emptyList()\n 1 -> listOf(this[0])\n else -> this.toMutableList()\n }\n}\n\n/**\n * Returns a [List] containing all elements.\n */\npublic fun ByteArray.toList(): List<Byte> {\n return when (size) {\n 0 -> emptyList()\n 1 -> listOf(this[0])\n else -> this.toMutableList()\n }\n}\n\n/**\n * Returns a [List] containing all elements.\n

```
*/\npublic fun ShortArray.toList(): List<Short> {\n    return when (size) {\n        0 -> emptyList()\n        1 -> listOf(this[0])\n        else -> this.toMutableList()\n    }\n}\n\n/**\n * Returns a [List] containing all elements.\n */\npublic fun IntArray.toList(): List<Int> {\n    return when (size) {\n        0 -> emptyList()\n        1 -> listOf(this[0])\n        else -> this.toMutableList()\n    }\n}\n\n/**\n * Returns a [List] containing all elements.\n */\npublic fun LongArray.toList(): List<Long> {\n    return when (size) {\n        0 -> emptyList()\n        1 -> listOf(this[0])\n        else -> this.toMutableList()\n    }\n}\n\n/**\n * Returns a [List] containing all elements.\n */\npublic fun FloatArray.toList(): List<Float> {\n    return when (size) {\n        0 -> emptyList()\n        1 -> listOf(this[0])\n        else -> this.toMutableList()\n    }\n}\n\n/**\n * Returns a [List] containing all elements.\n */\npublic fun DoubleArray.toList(): List<Double> {\n    return when (size) {\n        0 -> emptyList()\n        1 -> listOf(this[0])\n        else -> this.toMutableList()\n    }\n}\n\n/**\n * Returns a [List] containing all elements.\n */\npublic fun BooleanArray.toList(): List<Boolean> {\n    return when (size) {\n        0 -> emptyList()\n        1 -> listOf(this[0])\n        else -> this.toMutableList()\n    }\n}\n\n/**\n * Returns a [List] containing all elements.\n */\npublic fun CharArray.toList(): List<Char> {\n    return when (size) {\n        0 -> emptyList()\n        1 -> listOf(this[0])\n        else -> this.toMutableList()\n    }\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this array.\n */\npublic fun <T> Array<out T>.toMutableList(): MutableList<T> {\n    return ArrayList(this.asCollection())\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this array.\n */\npublic fun ByteArray.toMutableList(): MutableList<Byte> {\n    val list = ArrayList<Byte>(size)\n    for (item in this) list.add(item)\n    return list\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this array.\n */\npublic fun ShortArray.toMutableList(): MutableList<Short> {\n    val list = ArrayList<Short>(size)\n    for (item in this) list.add(item)\n    return list\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this array.\n */\npublic fun IntArray.toMutableList(): MutableList<Int> {\n    val list = ArrayList<Int>(size)\n    for (item in this) list.add(item)\n    return list\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this array.\n */\npublic fun LongArray.toMutableList(): MutableList<Long> {\n    val list = ArrayList<Long>(size)\n    for (item in this) list.add(item)\n    return list\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this array.\n */\npublic fun FloatArray.toMutableList(): MutableList<Float> {\n    val list = ArrayList<Float>(size)\n    for (item in this) list.add(item)\n    return list\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this array.\n */\npublic fun DoubleArray.toMutableList(): MutableList<Double> {\n    val list = ArrayList<Double>(size)\n    for (item in this) list.add(item)\n    return list\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this array.\n */\npublic fun BooleanArray.toMutableList(): MutableList<Boolean> {\n    val list = ArrayList<Boolean>(size)\n    for (item in this) list.add(item)\n    return list\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this array.\n */\npublic fun CharArray.toMutableList(): MutableList<Char> {\n    val list = ArrayList<Char>(size)\n    for (item in this) list.add(item)\n    return list\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun <T> Array<out T>.toSet(): Set<T> {\n    return when (size) {\n        0 -> emptySet()\n        1 -> setOf(this[0])\n        else -> toCollection(LinkedHashSet<T>(mapCapacity(size)))\n    }\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun ByteArray.toSet(): Set<Byte> {\n    return when (size) {\n        0 -> emptySet()\n        1 -> setOf(this[0])\n        else -> toCollection(LinkedHashSet<Byte>(mapCapacity(size)))\n    }\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun ShortArray.toSet(): Set<Short> {\n    return when (size) {\n        0 -> emptySet()\n        1 -> setOf(this[0])\n        else -> toCollection(LinkedHashSet<Short>(mapCapacity(size)))\n    }\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun IntArray.toSet(): Set<Int> {\n    return when (size) {\n        0 -> emptySet()\n        1 -> setOf(this[0])\n        else -> toCollection(LinkedHashSet<Int>(mapCapacity(size)))\n    }\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun LongArray.toSet(): Set<Long> {\n    return when (size) {\n        0 -> emptySet()\n        1 -> setOf(this[0])\n        else -> toCollection(LinkedHashSet<Long>(mapCapacity(size)))\n    }\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n
```

* The returned set preserves the element iteration order of the original array.\n */\npublic fun FloatArray.toSet(): Set<Float> {\n   return when (size) {\n   0 -> emptySet()\n   1 -> setOf(this[0])\n   else -> toCollection(LinkedHashSet<Float>(mapCapacity(size)))\n   }\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun DoubleArray.toSet(): Set<Double> {\n   return when (size) {\n   0 -> emptySet()\n   1 -> setOf(this[0])\n   else -> toCollection(LinkedHashSet<Double>(mapCapacity(size)))\n   }\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun BooleanArray.toSet(): Set<Boolean> {\n   return when (size) {\n   0 -> emptySet()\n   1 -> setOf(this[0])\n   else -> toCollection(LinkedHashSet<Boolean>(mapCapacity(size)))\n   }\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun CharArray.toSet(): Set<Char> {\n   return when (size) {\n   0 -> emptySet()\n   1 -> setOf(this[0])\n   else -> toCollection(LinkedHashSet<Char>(mapCapacity(size.coerceAtMost(128))))\n   }\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <T, R> Array<out T>.flatMap(transform: (T) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <R> ByteArray.flatMap(transform: (Byte) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <R> ShortArray.flatMap(transform: (Short) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <R> IntArray.flatMap(transform: (Int) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <R> LongArray.flatMap(transform: (Long) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <R> FloatArray.flatMap(transform: (Float) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <R> DoubleArray.flatMap(transform: (Double) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <R> BooleanArray.flatMap(transform: (Boolean) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <R> CharArray.flatMap(transform: (Char) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapSequence\")\npublic inline fun <T, R> Array<out

T>.flatMap(transform: (T) -> Sequence<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Array<out T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> ByteArray.flatMapIndexed(transform: (index: Int, Byte) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> ShortArray.flatMapIndexed(transform: (index: Int, Short) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> IntArray.flatMapIndexed(transform: (index: Int, Int) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> LongArray.flatMapIndexed(transform: (index: Int, Long) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> FloatArray.flatMapIndexed(transform: (index: Int, Float) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> DoubleArray.flatMapIndexed(transform: (index: Int, Double) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic

inline fun <R> BooleanArray.flatMapIndexed(transform: (index: Int, Boolean) -> Iterable<R>): List<R> {\n return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharArray.flatMapIndexed(transform: (index: Int, Char) -> Iterable<R>): List<R> {\n return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedSequence\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Array<out T>.flatMapIndexed(transform: (index: Int, T) -> Sequence<R>): List<R> {\n return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R, C : MutableCollection<in R>> Array<out T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(index++, element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> ByteArray.flatMapIndexedTo(destination: C, transform: (index: Int, Byte) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(index++, element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> ShortArray.flatMapIndexedTo(destination: C, transform: (index: Int, Short) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(index++, element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> IntArray.flatMapIndexedTo(destination: C, transform: (index: Int, Int) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(index++, element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> LongArray.flatMapIndexedTo(destination: C, transform: (index: Int, Long) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(index++,

element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> FloatArray.flatMapIndexedTo(destination: C, transform: (index: Int, Float) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(index++, element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> DoubleArray.flatMapIndexedTo(destination: C, transform: (index: Int, Double) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(index++, element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> BooleanArray.flatMapIndexedTo(destination: C, transform: (index: Int, Boolean) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(index++, element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> CharArray.flatMapIndexedTo(destination: C, transform: (index: Int, Char) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(index++, element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedSequenceTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R, C : MutableCollection<in R>> Array<out T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Sequence<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(index++, element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <T, R, C : MutableCollection<in R>> Array<out T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> ByteArray.flatMapTo(destination: C, transform: (Byte) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> ShortArray.flatMapTo(destination: C, transform: (Short) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given

[destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> IntArray.flatMapTo(destination: C, transform: (Int) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> LongArray.flatMapTo(destination: C, transform: (Long) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> FloatArray.flatMapTo(destination: C, transform: (Float) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> DoubleArray.flatMapTo(destination: C, transform: (Double) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapSequenceTo\")\npublic inline fun <T, R, C : MutableCollection<in R>> Array<out T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <T, K> Array<out T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {\n return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K> ByteArray.groupBy(keySelector: (Byte) -> K): Map<K, List<Byte>> {\n return groupByTo(LinkedHashMap<K, MutableList<Byte>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K> ShortArray.groupBy(keySelector: (Short) -> K): Map<K, List<Short>> {\n return groupByTo(LinkedHashMap<K, MutableList<Short>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K>

IntArray.groupBy(keySelector: (Int) -> K): Map<K, List<Int>> {\n   return groupByTo(LinkedHashMap<K, MutableList<Int>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K> LongArray.groupBy(keySelector: (Long) -> K): Map<K, List<Long>> {\n   return groupByTo(LinkedHashMap<K, MutableList<Long>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K> FloatArray.groupBy(keySelector: (Float) -> K): Map<K, List<Float>> {\n   return groupByTo(LinkedHashMap<K, MutableList<Float>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K> DoubleArray.groupBy(keySelector: (Double) -> K): Map<K, List<Double>> {\n   return groupByTo(LinkedHashMap<K, MutableList<Double>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K> BooleanArray.groupBy(keySelector: (Boolean) -> K): Map<K, List<Boolean>> {\n   return groupByTo(LinkedHashMap<K, MutableList<Boolean>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K> CharArray.groupBy(keySelector: (Char) -> K): Map<K, List<Char>> {\n   return groupByTo(LinkedHashMap<K, MutableList<Char>>(), keySelector)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <T, K, V> Array<out T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {\n   return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V> ByteArray.groupBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, List<V>> {\n   return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V>

ShortArray.groupBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, List<V>> {\n   return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V> IntArray.groupBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, List<V>> {\n   return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V> LongArray.groupBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, List<V>> {\n   return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V> FloatArray.groupBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, List<V>> {\n   return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V> DoubleArray.groupBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, List<V>> {\n   return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V> BooleanArray.groupBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, List<V>> {\n   return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V> CharArray.groupBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>> {\n   return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Array<out T>.groupByTo(destination: M, keySelector: (T) -

> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<T>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K, M : MutableMap<in K, MutableList<Byte>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<Byte>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K, M : MutableMap<in K, MutableList<Short>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<Short>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K, M : MutableMap<in K, MutableList<Int>>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<Int>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K, M : MutableMap<in K, MutableList<Long>>> LongArray.groupByTo(destination: M, keySelector: (Long) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<Long>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K, M : MutableMap<in K, MutableList<Float>>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<Float>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K, M : MutableMap<in K, MutableList<Double>>> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<Double>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K, M : MutableMap<in K, MutableList<Boolean>>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<Boolean>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K, M : MutableMap<in K,

MutableList<Char>>> CharArray.groupByTo(destination: M, keySelector: (Char) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<Char>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>> Array<out T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K, valueTransform: (Byte) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> LongArray.groupByTo(destination: M, keySelector: (Long) -> K, valueTransform: (Long) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K, valueTransform: (Float) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]

map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K, valueTransform: (Double) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> CharArray.groupByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Creates a [Grouping] source from an array to be used later with one of group-and-fold operations\n * using the specified [keySelector] function to extract a key from each element.\n * \n * @sample samples.collections.Grouping.groupingByEachCount\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K> Array<out T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {\n return object : Grouping<T, K> {\n override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()\n override fun keyOf(element: T): K = keySelector(element)\n }\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic inline fun <T, R> Array<out T>.map(transform: (T) -> R): List<R> {\n return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic inline fun <R> ByteArray.map(transform: (Byte) -> R): List<R> {\n return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic inline fun <R> ShortArray.map(transform: (Short) -> R): List<R> {\n return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic inline fun <R> IntArray.map(transform: (Int) -> R): List<R> {\n return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic inline fun <R> LongArray.map(transform: (Long) -> R): List<R> {\n return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic inline fun <R> FloatArray.map(transform: (Float) -> R): List<R> {\n return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic inline fun <R> DoubleArray.map(transform: (Double) -> R): List<R> {\n return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic inline fun <R> BooleanArray.map(transform:

(Boolean) -> R): List<R> {\n   return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic inline fun <R> CharArray.map(transform: (Char) -> R): List<R> {\n   return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <T, R> Array<out T>.mapIndexed(transform: (index: Int, T) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R> ByteArray.mapIndexed(transform: (index: Int, Byte) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R> ShortArray.mapIndexed(transform: (index: Int, Short) -> R): List<R> {\n return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R> IntArray.mapIndexed(transform: (index: Int, Int) -> R): List<R> {\n return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R> LongArray.mapIndexed(transform: (index: Int, Long) -> R): List<R> {\n return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R> FloatArray.mapIndexed(transform: (index: Int, Float) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R> DoubleArray.mapIndexed(transform: (index: Int, Double) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R> BooleanArray.mapIndexed(transform: (index: Int, Boolean) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R> CharArray.mapIndexed(transform: (index: Int, Char) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing only the non-null results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <T, R : Any> Array<out T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): List<R> {\n   return mapIndexedNotNullTo(ArrayList<R>(), transform)\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends only the non-null results to the given [destination].\n *

@param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <T, R : Any, C : MutableCollection<in R>> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {\n   forEachIndexed { index, element -> transform(index, element)?.let { destination.add(it) } }\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <T, R, C : MutableCollection<in R>> Array<out T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {\n   var index = 0\n   for (item in this)\n   destination.add(transform(index++, item))\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R, C : MutableCollection<in R>> ByteArray.mapIndexedTo(destination: C, transform: (index: Int, Byte) -> R): C {\n   var index = 0\n   for (item in this)\n   destination.add(transform(index++, item))\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R, C : MutableCollection<in R>> ShortArray.mapIndexedTo(destination: C, transform: (index: Int, Short) -> R): C {\n   var index = 0\n   for (item in this)\n   destination.add(transform(index++, item))\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R, C : MutableCollection<in R>> IntArray.mapIndexedTo(destination: C, transform: (index: Int, Int) -> R): C {\n   var index = 0\n   for (item in this)\n   destination.add(transform(index++, item))\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R, C : MutableCollection<in R>> LongArray.mapIndexedTo(destination: C, transform: (index: Int, Long) -> R): C {\n   var index = 0\n   for (item in this)\n   destination.add(transform(index++, item))\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R, C : MutableCollection<in R>> FloatArray.mapIndexedTo(destination: C, transform: (index: Int, Float) -> R): C {\n   var index = 0\n   for (item in this)\n   destination.add(transform(index++, item))\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R, C : MutableCollection<in R>> DoubleArray.mapIndexedTo(destination: C, transform: (index: Int, Double) -> R): C {\n   var index = 0\n   for (item in this)\n   destination.add(transform(index++, item))\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R, C : MutableCollection<in R>> BooleanArray.mapIndexedTo(destination: C, transform: (index: Int, Boolean) -> R): C {\n   var index = 0\n   for (item in this)\n   destination.add(transform(index++, item))\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <R, C : MutableCollection<in R>> CharArray.mapIndexedTo(destination: C, transform: (index: Int, Char) -> R): C {\n   var index = 0\n   for (item in

this)\n destination.add(transform(index++, item))\n return destination\n}\n\n/**\n * Returns a list containing only the non-null results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.mapNotNull\n */\npublic inline fun <T, R : Any> Array<out T>.mapNotNull(transform: (T) -> R?): List<R> {\n return mapNotNullTo(ArrayList<R>(), transform)\n}\n\n/**\n * Applies the given [transform] function to each element in the original array\n * and appends only the non-null results to the given [destination].\n */\npublic inline fun <T, R : Any, C : MutableCollection<in R>> Array<out T>.mapNotNullTo(destination: C, transform: (T) -> R?): C {\n forEach { element -> transform(element)?.let { destination.add(it) } }\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\npublic inline fun <T, R, C : MutableCollection<in R>> Array<out T>.mapTo(destination: C, transform: (T) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> ByteArray.mapTo(destination: C, transform: (Byte) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> ShortArray.mapTo(destination: C, transform: (Short) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> IntArray.mapTo(destination: C, transform: (Int) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> LongArray.mapTo(destination: C, transform: (Long) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> FloatArray.mapTo(destination: C, transform: (Float) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> DoubleArray.mapTo(destination: C, transform: (Double) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> BooleanArray.mapTo(destination: C, transform: (Boolean) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> CharArray.mapTo(destination: C, transform: (Char) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\npublic fun <T> Array<out T>.withIndex(): Iterable<IndexedValue<T>> {\n return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\npublic fun ByteArray.withIndex(): Iterable<IndexedValue<Byte>> {\n return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\npublic fun ShortArray.withIndex(): Iterable<IndexedValue<Short>> {\n return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\npublic fun IntArray.withIndex(): Iterable<IndexedValue<Int>> {\n return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\npublic

fun LongArray.withIndex(): Iterable<IndexedValue<Long>> {\n   return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\npublic fun FloatArray.withIndex(): Iterable<IndexedValue<Float>> {\n   return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\npublic fun DoubleArray.withIndex(): Iterable<IndexedValue<Double>> {\n   return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\npublic fun BooleanArray.withIndex(): Iterable<IndexedValue<Boolean>> {\n   return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\npublic fun CharArray.withIndex(): Iterable<IndexedValue<Char>> {\n   return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * Among equal elements of the given array, only the first one will be present in the resulting list.\n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun <T> Array<out T>.distinct(): List<T> {\n   return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun ByteArray.distinct(): List<Byte> {\n   return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun ShortArray.distinct(): List<Short> {\n   return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun IntArray.distinct(): List<Int> {\n   return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun LongArray.distinct(): List<Long> {\n   return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun FloatArray.distinct(): List<Float> {\n   return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun DoubleArray.distinct(): List<Double> {\n   return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun BooleanArray.distinct(): List<Boolean> {\n   return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun CharArray.distinct(): List<Char> {\n   return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * Among elements of the given array with equal keys, only the first one will be present in the resulting list.\n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic inline fun <T, K> Array<out

T>.distinctBy(selector: (T) -> K): List<T> {\n val set = HashSet<K>()\n val list = ArrayList<T>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic inline fun <K> ByteArray.distinctBy(selector: (Byte) -> K): List<Byte> {\n val set = HashSet<K>()\n val list = ArrayList<Byte>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic inline fun <K> ShortArray.distinctBy(selector: (Short) -> K): List<Short> {\n val set = HashSet<K>()\n val list = ArrayList<Short>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic inline fun <K> IntArray.distinctBy(selector: (Int) -> K): List<Int> {\n val set = HashSet<K>()\n val list = ArrayList<Int>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic inline fun <K> LongArray.distinctBy(selector: (Long) -> K): List<Long> {\n val set = HashSet<K>()\n val list = ArrayList<Long>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic inline fun <K> FloatArray.distinctBy(selector: (Float) -> K): List<Float> {\n val set = HashSet<K>()\n val list = ArrayList<Float>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic inline fun <K> DoubleArray.distinctBy(selector: (Double) -> K): List<Double> {\n val set = HashSet<K>()\n val list = ArrayList<Double>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic inline fun <K> BooleanArray.distinctBy(selector: (Boolean) -> K): List<Boolean> {\n val set = HashSet<K>()\n val list = ArrayList<Boolean>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic inline fun <K> CharArray.distinctBy(selector: (Char) -> K): List<Char> {\n val set = HashSet<K>()\n val list = ArrayList<Char>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n

return list\n}\n\n/**\n * Returns a set containing all elements that are contained by both this array and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n * \n * To get a set containing all elements that are contained at least in one of these collections use [union].\n */\npublic infix fun <T> Array<out T>.intersect(other: Iterable<T>): Set<T> {\n val set = this.toMutableSet()\n set.retainAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by both this array and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n * \n * To get a set containing all elements that are contained at least in one of these collections use [union].\n */\npublic infix fun ByteArray.intersect(other: Iterable<Byte>): Set<Byte> {\n val set = this.toMutableSet()\n set.retainAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by both this array and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n * \n * To get a set containing all elements that are contained at least in one of these collections use [union].\n */\npublic infix fun ShortArray.intersect(other: Iterable<Short>): Set<Short> {\n val set = this.toMutableSet()\n set.retainAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by both this array and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n * \n * To get a set containing all elements that are contained at least in one of these collections use [union].\n */\npublic infix fun IntArray.intersect(other: Iterable<Int>): Set<Int> {\n val set = this.toMutableSet()\n set.retainAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by both this array and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n * \n * To get a set containing all elements that are contained at least in one of these collections use [union].\n */\npublic infix fun LongArray.intersect(other: Iterable<Long>): Set<Long> {\n val set = this.toMutableSet()\n set.retainAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by both this array and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n * \n * To get a set containing all elements that are contained at least in one of these collections use [union].\n */\npublic infix fun FloatArray.intersect(other: Iterable<Float>): Set<Float> {\n val set = this.toMutableSet()\n set.retainAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by both this array and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n * \n * To get a set containing all elements that are contained at least in one of these collections use [union].\n */\npublic infix fun DoubleArray.intersect(other: Iterable<Double>): Set<Double> {\n val set = this.toMutableSet()\n set.retainAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by both this array and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n * \n * To get a set containing all elements that are contained at least in one of these collections use [union].\n */\npublic infix fun BooleanArray.intersect(other: Iterable<Boolean>): Set<Boolean> {\n val set = this.toMutableSet()\n set.retainAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by both this array and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n * \n * To get a set containing all elements that are contained at least in one of these collections use [union].\n */\npublic infix fun CharArray.intersect(other: Iterable<Char>): Set<Char> {\n val set = this.toMutableSet()\n set.retainAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun <T> Array<out T>.subtract(other: Iterable<T>): Set<T> {\n val set = this.toMutableSet()\n set.removeAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> {\n val set = this.toMutableSet()\n set.removeAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> {\n val set = this.toMutableSet()\n set.removeAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the

element iteration order of the original array.\n */\npublic infix fun IntArray.subtract(other: Iterable<Int>): Set<Int> {\n   val set = this.toMutableSet()\n   set.removeAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun LongArray.subtract(other: Iterable<Long>): Set<Long> {\n   val set = this.toMutableSet()\n   set.removeAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun FloatArray.subtract(other: Iterable<Float>): Set<Float> {\n   val set = this.toMutableSet()\n   set.removeAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun DoubleArray.subtract(other: Iterable<Double>): Set<Double> {\n   val set = this.toMutableSet()\n   set.removeAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun BooleanArray.subtract(other: Iterable<Boolean>): Set<Boolean> {\n   val set = this.toMutableSet()\n   set.removeAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> {\n   val set = this.toMutableSet()\n   set.removeAll(other)\n   return set\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given array.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun <T> Array<out T>.toMutableSet(): MutableSet<T> {\n   return toCollection(LinkedHashSet<T>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given array.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun ByteArray.toMutableSet(): MutableSet<Byte> {\n   return toCollection(LinkedHashSet<Byte>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given array.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun ShortArray.toMutableSet(): MutableSet<Short> {\n   return toCollection(LinkedHashSet<Short>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given array.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun IntArray.toMutableSet(): MutableSet<Int> {\n   return toCollection(LinkedHashSet<Int>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given array.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun LongArray.toMutableSet(): MutableSet<Long> {\n   return toCollection(LinkedHashSet<Long>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given array.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun FloatArray.toMutableSet(): MutableSet<Float> {\n   return toCollection(LinkedHashSet<Float>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given array.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun DoubleArray.toMutableSet(): MutableSet<Double> {\n   return toCollection(LinkedHashSet<Double>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given array.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun BooleanArray.toMutableSet(): MutableSet<Boolean> {\n   return toCollection(LinkedHashSet<Boolean>(mapCapacity(size)))\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given array.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic fun CharArray.toMutableSet(): MutableSet<Char> {\n   return toCollection(LinkedHashSet<Char>(mapCapacity(size.coerceAtMost(128))))\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n * \n * To get a set containing all elements that are contained in both collections use

[intersect].\n */\npublic infix fun <T> Array<out T>.union(other: Iterable<T>): Set<T> {\n   val set = this.toMutableSet()\n   set.addAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n * \n * To get a set containing all elements that are contained in both collections use [intersect].\n */\npublic infix fun ByteArray.union(other: Iterable<Byte>): Set<Byte> {\n   val set = this.toMutableSet()\n   set.addAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n * \n * To get a set containing all elements that are contained in both collections use [intersect].\n */\npublic infix fun ShortArray.union(other: Iterable<Short>): Set<Short> {\n   val set = this.toMutableSet()\n   set.addAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n * \n * To get a set containing all elements that are contained in both collections use [intersect].\n */\npublic infix fun IntArray.union(other: Iterable<Int>): Set<Int> {\n   val set = this.toMutableSet()\n   set.addAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n * \n * To get a set containing all elements that are contained in both collections use [intersect].\n */\npublic infix fun LongArray.union(other: Iterable<Long>): Set<Long> {\n   val set = this.toMutableSet()\n   set.addAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n * \n * To get a set containing all elements that are contained in both collections use [intersect].\n */\npublic infix fun FloatArray.union(other: Iterable<Float>): Set<Float> {\n   val set = this.toMutableSet()\n   set.addAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n * \n * To get a set containing all elements that are contained in both collections use [intersect].\n */\npublic infix fun DoubleArray.union(other: Iterable<Double>): Set<Double> {\n   val set = this.toMutableSet()\n   set.addAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n * \n * To get a set containing all elements that are contained in both collections use [intersect].\n */\npublic infix fun BooleanArray.union(other: Iterable<Boolean>): Set<Boolean> {\n   val set = this.toMutableSet()\n   set.addAll(other)\n   return set\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n * \n * To get a set containing all elements that are contained in both collections use [intersect].\n */\npublic infix fun CharArray.union(other: Iterable<Char>): Set<Char> {\n   val set = this.toMutableSet()\n   set.addAll(other)\n   return set\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun <T> Array<out T>.all(predicate: (T) -> Boolean): Boolean {\n   for (element in this) if (!predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun ByteArray.all(predicate: (Byte) -> Boolean): Boolean {\n   for (element in this) if (!predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun ShortArray.all(predicate: (Short) -> Boolean): Boolean {\n   for (element in this) if (!predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if all elements match the given

[predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun IntArray.all(predicate: (Int) -> Boolean): Boolean {\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun LongArray.all(predicate: (Long) -> Boolean): Boolean {\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun FloatArray.all(predicate: (Float) -> Boolean): Boolean {\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun DoubleArray.all(predicate: (Double) -> Boolean): Boolean {\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun BooleanArray.all(predicate: (Boolean) -> Boolean): Boolean {\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun CharArray.all(predicate: (Char) -> Boolean): Boolean {\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun <T> Array<out T>.any(): Boolean {\n return !isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun ByteArray.any(): Boolean {\n return !isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun ShortArray.any(): Boolean {\n return !isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun IntArray.any(): Boolean {\n return !isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun LongArray.any(): Boolean {\n return !isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun FloatArray.any(): Boolean {\n return !isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun DoubleArray.any(): Boolean {\n return !isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun BooleanArray.any(): Boolean {\n return !isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun CharArray.any(): Boolean {\n return !isEmpty()\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun <T> Array<out T>.any(predicate: (T) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun ByteArray.any(predicate: (Byte) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun ShortArray.any(predicate: (Short) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun IntArray.any(predicate: (Int) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun LongArray.any(predicate: (Long) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return

false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun FloatArray.any(predicate: (Float) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun DoubleArray.any(predicate: (Double) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun BooleanArray.any(predicate: (Boolean) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun CharArray.any(predicate: (Char) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns the number of elements in this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.count(): Int {\n return size\n}\n\n/**\n * Returns the number of elements in this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.count(): Int {\n return size\n}\n\n/**\n * Returns the number of elements in this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.count(): Int {\n return size\n}\n\n/**\n * Returns the number of elements in this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.count(): Int {\n return size\n}\n\n/**\n * Returns the number of elements in this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.count(): Int {\n return size\n}\n\n/**\n * Returns the number of elements in this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.count(): Int {\n return size\n}\n\n/**\n * Returns the number of elements in this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.count(): Int {\n return size\n}\n\n/**\n * Returns the number of elements in this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.count(): Int {\n return size\n}\n\n/**\n * Returns the number of elements in this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.count(): Int {\n return size\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun <T> Array<out T>.count(predicate: (T) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun ByteArray.count(predicate: (Byte) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun ShortArray.count(predicate: (Short) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun IntArray.count(predicate: (Int) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun LongArray.count(predicate: (Long) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun FloatArray.count(predicate: (Float) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun DoubleArray.count(predicate: (Double) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun BooleanArray.count(predicate: (Boolean) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun CharArray.count(predicate: (Char) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <T, R> Array<out

T>.fold(initial: R, operation: (acc: R, T) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> ByteArray.fold(initial: R, operation: (acc: R, Byte) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> ShortArray.fold(initial: R, operation: (acc: R, Short) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> IntArray.fold(initial: R, operation: (acc: R, Int) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> LongArray.fold(initial: R, operation: (acc: R, Long) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> FloatArray.fold(initial: R, operation: (acc: R, Float) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> DoubleArray.fold(initial: R, operation: (acc: R, Double) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> BooleanArray.fold(initial: R, operation: (acc: R, Boolean) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> CharArray.fold(initial: R, operation: (acc: R, Char) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <T, R> Array<out T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {\n   var index = 0\n   var accumulator = initial\n   for (element in this) accumulator = operation(index++, accumulator, element)\n   return

accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <R> ByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): R {\n var index = 0\n var accumulator = initial\n for (element in this) accumulator = operation(index++, accumulator, element)\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <R> ShortArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): R {\n var index = 0\n var accumulator = initial\n for (element in this) accumulator = operation(index++, accumulator, element)\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): R {\n var index = 0\n var accumulator = initial\n for (element in this) accumulator = operation(index++, accumulator, element)\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <R> LongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Long) -> R): R {\n var index = 0\n var accumulator = initial\n for (element in this) accumulator = operation(index++, accumulator, element)\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <R> FloatArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): R {\n var index = 0\n var accumulator = initial\n for (element in this) accumulator = operation(index++, accumulator, element)\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <R> DoubleArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): R {\n var index = 0\n var accumulator = initial\n for (element in this) accumulator = operation(index++, accumulator, element)\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <R> BooleanArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): R {\n var index = 0\n var accumulator = initial\n for (element in this) accumulator = operation(index++, accumulator, element)\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <R> CharArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R

{\n   var index = 0\n   var accumulator = initial\n   for (element in this) accumulator = operation(index++, accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <T, R> Array<out T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> ByteArray.foldRight(initial: R, operation: (Byte, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> ShortArray.foldRight(initial: R, operation: (Short, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> IntArray.foldRight(initial: R, operation: (Int, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> LongArray.foldRight(initial: R, operation: (Long, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> FloatArray.foldRight(initial: R, operation: (Float, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> DoubleArray.foldRight(initial: R, operation: (Double, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> BooleanArray.foldRight(initial: R, operation: (Boolean, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element

and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> CharArray.foldRight(initial: R, operation: (Char, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <T, R> Array<out T>.foldRightIndexed(initial: R, operation: (index: Int, T, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> ByteArray.foldRightIndexed(initial: R, operation: (index: Int, Byte, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> ShortArray.foldRightIndexed(initial: R, operation: (index: Int, Short, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> IntArray.foldRightIndexed(initial: R, operation: (index: Int, Int, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> LongArray.foldRightIndexed(initial: R, operation: (index: Int, Long, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> FloatArray.foldRightIndexed(initial: R, operation: (index: Int, Float, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> DoubleArray.foldRightIndexed(initial: R, operation: (index: Int, Double, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return

accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> BooleanArray.foldRightIndexed(initial: R, operation: (index: Int, Boolean, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> CharArray.foldRightIndexed(initial: R, operation: (index: Int, Char, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun <T> Array<out T>.forEach(action: (T) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun ByteArray.forEach(action: (Byte) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun ShortArray.forEach(action: (Short) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun IntArray.forEach(action: (Int) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun LongArray.forEach(action: (Long) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun FloatArray.forEach(action: (Float) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun DoubleArray.forEach(action: (Double) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun BooleanArray.forEach(action: (Boolean) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun CharArray.forEach(action: (Char) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\npublic inline fun <T> Array<out T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {\n var index = 0\n for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\npublic inline fun ByteArray.forEachIndexed(action: (index: Int, Byte) -> Unit): Unit {\n var index = 0\n for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\npublic inline fun ShortArray.forEachIndexed(action: (index: Int, Short) -> Unit): Unit {\n var index = 0\n for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\npublic inline fun IntArray.forEachIndexed(action: (index: Int, Int) -> Unit): Unit {\n var index = 0\n for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\npublic inline fun LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {\n var index = 0\n for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\npublic inline fun FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {\n var index = 0\n for (item in this)

action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\npublic inline fun DoubleArray.forEachIndexed(action: (index: Int, Double) -> Unit): Unit {\n var index = 0\n for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\npublic inline fun BooleanArray.forEachIndexed(action: (index: Int, Boolean) -> Unit): Unit {\n var index = 0\n for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\npublic inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {\n var index = 0\n for (item in this) action(index++, item)\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Array<out Double>.max(): Double? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Array<out Float>.max(): Float? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T : Comparable<T>> Array<out T>.max(): T? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun ByteArray.max(): Byte? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun ShortArray.max(): Short? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun IntArray.max(): Int? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun LongArray.max(): Long? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun FloatArray.max(): Float? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun DoubleArray.max(): Double? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun CharArray.max(): Char? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <T, R : Comparable<R>> Array<out T>.maxBy(selector: (T) -> R): T? {\n return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> ByteArray.maxBy(selector: (Byte) -> R): Byte? {\n return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> ShortArray.maxBy(selector: (Short) -> R): Short? {\n return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> IntArray.maxBy(selector: (Int) -> R): Int?

{\n   return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> LongArray.maxBy(selector: (Long) -> R): Long? {\n   return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> FloatArray.maxBy(selector: (Float) -> R): Float? {\n   return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> DoubleArray.maxBy(selector: (Double) -> R): Double? {\n   return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> BooleanArray.maxBy(selector: (Boolean) -> R): Boolean? {\n   return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> CharArray.maxBy(selector: (Char) -> R): Char? {\n   return maxByOrNull(selector)\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, R : Comparable<R>> Array<out T>.maxByOrNull(selector: (T) -> R): T? {\n   if (isEmpty()) return null\n   var maxElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return maxElem\n   var maxValue = selector(maxElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (maxValue < v) {\n   maxElem = e\n   maxValue = v\n   }\n   }\n   return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> ByteArray.maxByOrNull(selector: (Byte) -> R): Byte? {\n   if (isEmpty()) return null\n   var maxElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return maxElem\n   var maxValue = selector(maxElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (maxValue < v) {\n   maxElem = e\n   maxValue = v\n   }\n   }\n   return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> ShortArray.maxByOrNull(selector: (Short) -> R): Short? {\n   if (isEmpty()) return null\n   var maxElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return maxElem\n   var maxValue = selector(maxElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (maxValue < v) {\n   maxElem = e\n   maxValue = v\n   }\n   }\n   return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> IntArray.maxByOrNull(selector: (Int) -> R): Int? {\n   if (isEmpty()) return null\n   var maxElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return maxElem\n   var maxValue = selector(maxElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (maxValue < v) {\n   maxElem = e\n   maxValue = v\n   }\n   }\n   return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> LongArray.maxByOrNull(selector: (Long) -> R): Long? {\n   if (isEmpty()) return null\n   var maxElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return maxElem\n   var maxValue = selector(maxElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (maxValue < v) {\n   maxElem = e\n   maxValue = v\n   }\n   }\n   return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R :

Comparable<R>> FloatArray.maxByOrNull(selector: (Float) -> R): Float? {\n if (isEmpty()) return null\n var maxElem = this[0]\n val lastIndex = this.lastIndex\n if (lastIndex == 0) return maxElem\n var maxValue = selector(maxElem)\n for (i in 1..lastIndex) {\n val e = this[i]\n val v = selector(e)\n if (maxValue < v) {\n maxElem = e\n maxValue = v\n }\n }\n return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> DoubleArray.maxByOrNull(selector: (Double) -> R): Double? {\n if (isEmpty()) return null\n var maxElem = this[0]\n val lastIndex = this.lastIndex\n if (lastIndex == 0) return maxElem\n var maxValue = selector(maxElem)\n for (i in 1..lastIndex) {\n val e = this[i]\n val v = selector(e)\n if (maxValue < v) {\n maxElem = e\n maxValue = v\n }\n }\n return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> BooleanArray.maxByOrNull(selector: (Boolean) -> R): Boolean? {\n if (isEmpty()) return null\n var maxElem = this[0]\n val lastIndex = this.lastIndex\n if (lastIndex == 0) return maxElem\n var maxValue = selector(maxElem)\n for (i in 1..lastIndex) {\n val e = this[i]\n val v = selector(e)\n if (maxValue < v) {\n maxElem = e\n maxValue = v\n }\n }\n return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> CharArray.maxByOrNull(selector: (Char) -> R): Char? {\n if (isEmpty()) return null\n var maxElem = this[0]\n val lastIndex = this.lastIndex\n if (lastIndex == 0) return maxElem\n var maxValue = selector(maxElem)\n for (i in 1..lastIndex) {\n val e = this[i]\n val v = selector(e)\n if (maxValue < v) {\n maxElem = e\n maxValue = v\n }\n }\n return maxElem\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.maxOf(selector: (T) -> Double): Double {\n if (isEmpty()) throw NoSuchElementException()\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.maxOf(selector: (Byte) -> Double): Double {\n if (isEmpty()) throw NoSuchElementException()\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.maxOf(selector: (Short) -> Double): Double {\n if (isEmpty()) throw NoSuchElementException()\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.maxOf(selector: (Int) -> Double):

Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.maxOf(selector: (Long) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.maxOf(selector: (Float) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.maxOf(selector: (Double) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.maxOf(selector: (Boolean) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.maxOf(selector: (Char) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.maxOf(selector: (T) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.maxOf(selector: (Byte) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.maxOf(selector: (Short) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.maxOf(selector: (Int) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.maxOf(selector: (Long) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.maxOf(selector: (Float) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.maxOf(selector: (Double) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.maxOf(selector: (Boolean) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n

*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.maxOf(selector: (Char) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Array<out T>.maxOf(selector: (T) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ByteArray.maxOf(selector: (Byte) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ShortArray.maxOf(selector: (Short) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> IntArray.maxOf(selector: (Int) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> LongArray.maxOf(selector: (Long) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> FloatArray.maxOf(selector: (Float) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n

*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> DoubleArray.maxOf(selector: (Double) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> BooleanArray.maxOf(selector: (Boolean) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> CharArray.maxOf(selector: (Char) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.maxOfOrNull(selector: (T) -> Double): Double? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.maxOfOrNull(selector: (Byte) -> Double): Double? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.maxOfOrNull(selector: (Short) -> Double): Double? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.maxOfOrNull(selector: (Int) -> Double): Double? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is

`NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.maxOfOrNull(selector: (Long) -
> Double): Double? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex)
{\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n *
Returns the largest value among all values produced by [selector] function\n * applied to each element in the array
or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned
result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.maxOfOrNull(selector: (Float) -
> Double): Double? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex)
{\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n *
Returns the largest value among all values produced by [selector] function\n * applied to each element in the array
or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned
result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.maxOfOrNull(selector:
(Double) -> Double): Double? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in
1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return
maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to
each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function
is `NaN`, the returned result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.maxOfOrNull(selector:
(Boolean) -> Double): Double? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in
1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return
maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to
each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function
is `NaN`, the returned result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.maxOfOrNull(selector: (Char) ->
Double): Double? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n
    val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns
the largest value among all values produced by [selector] function\n * applied to each element in the array or `null`
if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is
`NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.maxOfOrNull(selector:
(T) -> Float): Float? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex)
{\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n *
Returns the largest value among all values produced by [selector] function\n * applied to each element in the array
or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned
result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.maxOfOrNull(selector: (Byte) ->
Float): Float? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n
val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns
the largest value among all values produced by [selector] function\n * applied to each element in the array or `null`

if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.maxOfOrNull(selector: (Short) -> Float): Float? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.maxOfOrNull(selector: (Int) -> Float): Float? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.maxOfOrNull(selector: (Long) -> Float): Float? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.maxOfOrNull(selector: (Float) -> Float): Float? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.maxOfOrNull(selector: (Double) -> Float): Float? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.maxOfOrNull(selector: (Boolean) -> Float): Float? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.maxOfOrNull(selector: (Char) -> Float): Float? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns

the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Array<out T>.maxOfOrNull(selector: (T) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ByteArray.maxOfOrNull(selector: (Byte) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ShortArray.maxOfOrNull(selector: (Short) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> IntArray.maxOfOrNull(selector: (Int) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> LongArray.maxOfOrNull(selector: (Long) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> FloatArray.maxOfOrNull(selector: (Float) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> DoubleArray.maxOfOrNull(selector: (Double) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> BooleanArray.maxOfOrNull(selector: (Boolean) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n

maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> CharArray.maxOfOrNull(selector: (Char) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Array<out T>.maxOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> ByteArray.maxOfWith(comparator: Comparator<in R>, selector: (Byte) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> ShortArray.maxOfWith(comparator: Comparator<in R>, selector: (Short) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> IntArray.maxOfWith(comparator: Comparator<in R>, selector: (Int) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> LongArray.maxOfWith(comparator: Comparator<in R>, selector: (Long) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> FloatArray.maxOfWith(comparator: Comparator<in R>, selector: (Float) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var

maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> DoubleArray.maxOfWith(comparator: Comparator<in R>, selector: (Double) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> BooleanArray.maxOfWith(comparator: Comparator<in R>, selector: (Boolean) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharArray.maxOfWith(comparator: Comparator<in R>, selector: (Char) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Array<out T>.maxOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> ByteArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Byte) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> ShortArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Short) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R>

IntArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Int) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> LongArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Long) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> FloatArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Float) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> DoubleArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Double) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> BooleanArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Boolean) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun Array<out Double>.maxOrNull(): Double? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   max = maxOf(max, e)\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun Array<out Float>.maxOrNull(): Float? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   max = maxOf(max, e)\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T : Comparable<T>> Array<out T>.maxOrNull(): T? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (max < e) max = e\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n

*/\n@SinceKotlin(\"1.4\")\npublic fun ByteArray.maxOrNull(): Byte? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (max < e) max = e\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun ShortArray.maxOrNull(): Short? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (max < e) max = e\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun IntArray.maxOrNull(): Int? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (max < e) max = e\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun LongArray.maxOrNull(): Long? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (max < e) max = e\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun FloatArray.maxOrNull(): Float? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   max = maxOf(max, e)\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun DoubleArray.maxOrNull(): Double? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   max = maxOf(max, e)\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun CharArray.maxOrNull(): Char? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (max < e) max = e\n   }\n   return max\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T> Array<out T>.maxWith(comparator: Comparator<in T>): T? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun ByteArray.maxWith(comparator: Comparator<in Byte>): Byte? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun ShortArray.maxWith(comparator: Comparator<in Short>): Short? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun IntArray.maxWith(comparator: Comparator<in Int>): Int? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun LongArray.maxWith(comparator: Comparator<in Long>): Long? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun FloatArray.maxWith(comparator: Comparator<in Float>): Float? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun DoubleArray.maxWith(comparator: Comparator<in Double>): Double? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun BooleanArray.maxWith(comparator: Comparator<in Boolean>): Boolean? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun CharArray.maxWith(comparator: Comparator<in Char>): Char? {\n   return maxWithOrNull(comparator)\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Array<out

T>.maxWithOrNull(comparator: Comparator<in T>): T? {\n if (isEmpty()) return null\n var max = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun ByteArray.maxWithOrNull(comparator: Comparator<in Byte>): Byte? {\n if (isEmpty()) return null\n var max = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun ShortArray.maxWithOrNull(comparator: Comparator<in Short>): Short? {\n if (isEmpty()) return null\n var max = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun IntArray.maxWithOrNull(comparator: Comparator<in Int>): Int? {\n if (isEmpty()) return null\n var max = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun LongArray.maxWithOrNull(comparator: Comparator<in Long>): Long? {\n if (isEmpty()) return null\n var max = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun FloatArray.maxWithOrNull(comparator: Comparator<in Float>): Float? {\n if (isEmpty()) return null\n var max = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun DoubleArray.maxWithOrNull(comparator: Comparator<in Double>): Double? {\n if (isEmpty()) return null\n var max = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun BooleanArray.maxWithOrNull(comparator: Comparator<in Boolean>): Boolean? {\n if (isEmpty()) return null\n var max = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun CharArray.maxWithOrNull(comparator: Comparator<in Char>): Char? {\n if (isEmpty()) return null\n var max = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Array<out Double>.min(): Double? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Array<out Float>.min(): Float? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T : Comparable<T>> Array<out T>.min(): T? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun ByteArray.min(): Byte? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun ShortArray.min(): Short? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\",

ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun IntArray.min(): Int? {\n   return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun LongArray.min(): Long? {\n   return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun FloatArray.min(): Float? {\n   return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun DoubleArray.min(): Double? {\n   return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun CharArray.min(): Char? {\n   return minOrNull()\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <T, R : Comparable<R>> Array<out T>.minBy(selector: (T) -> R): T? {\n   return minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> ByteArray.minBy(selector: (Byte) -> R): Byte? {\n   return minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> ShortArray.minBy(selector: (Short) -> R): Short? {\n   return minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> IntArray.minBy(selector: (Int) -> R): Int? {\n   return minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> LongArray.minBy(selector: (Long) -> R): Long? {\n   return minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> FloatArray.minBy(selector: (Float) -> R): Float? {\n   return minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> DoubleArray.minBy(selector: (Double) -> R): Double? {\n   return minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> BooleanArray.minBy(selector: (Boolean) -> R): Boolean? {\n   return minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> CharArray.minBy(selector: (Char) -> R): Char? {\n   return minByOrNull(selector)\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, R : Comparable<R>> Array<out T>.minByOrNull(selector: (T) -> R): T? {\n   if (isEmpty()) return null\n   var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue = selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v) {\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> ByteArray.minByOrNull(selector: (Byte) -> R): Byte? {\n   if (isEmpty()) return null\n   var

minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue = selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v) {\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> ShortArray.minByOrNull(selector: (Short) -> R): Short? {\n   if (isEmpty()) return null\n   var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue = selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v) {\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> IntArray.minByOrNull(selector: (Int) -> R): Int? {\n   if (isEmpty()) return null\n   var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue = selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v) {\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> LongArray.minByOrNull(selector: (Long) -> R): Long? {\n   if (isEmpty()) return null\n   var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue = selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v) {\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> FloatArray.minByOrNull(selector: (Float) -> R): Float? {\n   if (isEmpty()) return null\n   var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue = selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v) {\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> DoubleArray.minByOrNull(selector: (Double) -> R): Double? {\n   if (isEmpty()) return null\n   var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue = selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v) {\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> BooleanArray.minByOrNull(selector: (Boolean) -> R): Boolean? {\n   if (isEmpty()) return null\n   var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue = selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v) {\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> CharArray.minByOrNull(selector: (Char) -> R): Char? {\n   if (isEmpty()) return null\n   var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue = selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v) {\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n

*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.minOf(selector: (T) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.minOf(selector: (Byte) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.minOf(selector: (Short) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.minOf(selector: (Int) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.minOf(selector: (Long) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.minOf(selector: (Float) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.minOf(selector: (Double) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is

`NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.minOf(selector: (Boolean) -> Double): Double {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.minOf(selector: (Char) -> Double): Double {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.minOf(selector: (T) -> Float): Float {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.minOf(selector: (Byte) -> Float): Float {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.minOf(selector: (Short) -> Float): Float {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.minOf(selector: (Int) -> Float): Float {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.minOf(selector: (Long) -> Float): Float {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to

each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.minOf(selector: (Float) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.minOf(selector: (Double) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.minOf(selector: (Boolean) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.minOf(selector: (Char) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Array<out T>.minOf(selector: (T) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ByteArray.minOf(selector: (Byte) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ShortArray.minOf(selector: (Short) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values

produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> IntArray.minOf(selector: (Int) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> LongArray.minOf(selector: (Long) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> FloatArray.minOf(selector: (Float) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> DoubleArray.minOf(selector: (Double) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> BooleanArray.minOf(selector: (Boolean) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> CharArray.minOf(selector: (Char) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.minOfOrNull(selector: (T) -> Double): Double? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return

minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.minOfOrNull(selector: (Byte) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.minOfOrNull(selector: (Short) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.minOfOrNull(selector: (Int) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.minOfOrNull(selector: (Long) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.minOfOrNull(selector: (Float) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.minOfOrNull(selector: (Double) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.minOfOrNull(selector: (Boolean) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in

1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.minOfOrNull(selector: (Char) -> Double): Double? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.minOfOrNull(selector: (T) -> Float): Float? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.minOfOrNull(selector: (Byte) -> Float): Float? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.minOfOrNull(selector: (Short) -> Float): Float? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.minOfOrNull(selector: (Int) -> Float): Float? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.minOfOrNull(selector: (Long) -> Float): Float? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.minOfOrNull(selector: (Float) ->

Float): Float? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.minOfOrNull(selector: (Double) -> Float): Float? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.minOfOrNull(selector: (Boolean) -> Float): Float? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.minOfOrNull(selector: (Char) -> Float): Float? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Array<out T>.minOfOrNull(selector: (T) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ByteArray.minOfOrNull(selector: (Byte) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ShortArray.minOfOrNull(selector: (Short) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> IntArray.minOfOrNull(selector: (Int) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced

by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> LongArray.minOfOrNull(selector: (Long) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> FloatArray.minOfOrNull(selector: (Float) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> DoubleArray.minOfOrNull(selector: (Double) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> BooleanArray.minOfOrNull(selector: (Boolean) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> CharArray.minOfOrNull(selector: (Char) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Array<out T>.minOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> ByteArray.minOfWith(comparator: Comparator<in R>, selector: (Byte) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType

ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> ShortArray.minOfWith(comparator: Comparator<in R>, selector: (Short) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> IntArray.minOfWith(comparator: Comparator<in R>, selector: (Int) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> LongArray.minOfWith(comparator: Comparator<in R>, selector: (Long) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> FloatArray.minOfWith(comparator: Comparator<in R>, selector: (Float) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> DoubleArray.minOfWith(comparator: Comparator<in R>, selector: (Double) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> BooleanArray.minOfWith(comparator: Comparator<in R>, selector: (Boolean) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharArray.minOfWith(comparator: Comparator<in R>, selector: (Char) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n

*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Array<out T>.minOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> ByteArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Byte) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> ShortArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Short) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> IntArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Int) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> LongArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Long) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> FloatArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Float) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> DoubleArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Double) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector]

function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> BooleanArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Boolean) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun Array<out Double>.minOrNull(): Double? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n min = minOf(min, e)\n }\n return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun Array<out Float>.minOrNull(): Float? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n min = minOf(min, e)\n }\n return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T : Comparable<T>> Array<out T>.minOrNull(): T? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (min > e) min = e\n }\n return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun ByteArray.minOrNull(): Byte? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (min > e) min = e\n }\n return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun ShortArray.minOrNull(): Short? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (min > e) min = e\n }\n return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun IntArray.minOrNull(): Int? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (min > e) min = e\n }\n return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun LongArray.minOrNull(): Long? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (min > e) min = e\n }\n return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun FloatArray.minOrNull(): Float? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n min = minOf(min, e)\n }\n return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun DoubleArray.minOrNull(): Double? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n min = minOf(min, e)\n }\n return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun CharArray.minOrNull(): Char? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (min > e) min = e\n }\n return min\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T> Array<out T>.minWith(comparator: Comparator<in T>): T? {\n return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun ByteArray.minWith(comparator: Comparator<in Byte>): Byte? {\n

return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun ShortArray.minWith(comparator: Comparator<in Short>): Short? {\n return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun IntArray.minWith(comparator: Comparator<in Int>): Int? {\n return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun LongArray.minWith(comparator: Comparator<in Long>): Long? {\n return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun FloatArray.minWith(comparator: Comparator<in Float>): Float? {\n return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun DoubleArray.minWith(comparator: Comparator<in Double>): Double? {\n return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun BooleanArray.minWith(comparator: Comparator<in Boolean>): Boolean? {\n return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun CharArray.minWith(comparator: Comparator<in Char>): Char? {\n return minWithOrNull(comparator)\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Array<out T>.minWithOrNull(comparator: Comparator<in T>): T? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun ByteArray.minWithOrNull(comparator: Comparator<in Byte>): Byte? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun ShortArray.minWithOrNull(comparator: Comparator<in Short>): Short? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun IntArray.minWithOrNull(comparator: Comparator<in Int>): Int? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun LongArray.minWithOrNull(comparator: Comparator<in Long>): Long? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun FloatArray.minWithOrNull(comparator: Comparator<in Float>): Float? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun DoubleArray.minWithOrNull(comparator: Comparator<in Double>): Double? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if

(comparator.compare(min, e) > 0) min = e\n   }\n   return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun BooleanArray.minWithOrNull(comparator: Comparator<in Boolean>): Boolean? {\n   if (isEmpty()) return null\n   var min = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(min, e) > 0) min = e\n   }\n   return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun CharArray.minWithOrNull(comparator: Comparator<in Char>): Char? {\n   if (isEmpty()) return null\n   var min = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(min, e) > 0) min = e\n   }\n   return min\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun <T> Array<out T>.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun ByteArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun ShortArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun IntArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun LongArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun FloatArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun DoubleArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun BooleanArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun CharArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun <T> Array<out T>.none(predicate: (T) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun ByteArray.none(predicate: (Byte) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun ShortArray.none(predicate: (Short) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun IntArray.none(predicate: (Int) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun LongArray.none(predicate: (Long) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun FloatArray.none(predicate: (Float) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun DoubleArray.none(predicate: (Double) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample

samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun BooleanArray.none(predicate: (Boolean) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun CharArray.none(predicate: (Char) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.onEach(action: (T) -> Unit): Array<out T> {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.onEach(action: (Byte) -> Unit): ByteArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.onEach(action: (Short) -> Unit): ShortArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.onEach(action: (Int) -> Unit): IntArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.onEach(action: (Long) -> Unit): LongArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.onEach(action: (Float) -> Unit): FloatArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.onEach(action: (Double) -> Unit): DoubleArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.onEach(action: (Boolean) -> Unit): BooleanArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.onEach(action: (Char) -> Unit): CharArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.onEachIndexed(action: (index: Int, T) -> Unit): Array<out T> {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.onEachIndexed(action: (index: Int, Byte) -> Unit): ByteArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.onEachIndexed(action: (index: Int, Short) -> Unit): ShortArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun

IntArray.onEachIndexed(action: (index: Int, Int) -> Unit): IntArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.onEachIndexed(action: (index: Int, Long) -> Unit): LongArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.onEachIndexed(action: (index: Int, Float) -> Unit): FloatArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.onEachIndexed(action: (index: Int, Double) -> Unit): DoubleArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.onEachIndexed(action: (index: Int, Boolean) -> Unit): BooleanArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.onEachIndexed(action: (index: Int, Char) -> Unit): CharArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun <S, T : S> Array<out T>.reduce(operation: (acc: S, T) -> S): S {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator: S = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun ByteArray.reduce(operation: (acc: Byte, Byte) -> Byte): Byte {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun ShortArray.reduce(operation: (acc: Short, Short) -> Short): Short {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current

accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun IntArray.reduce(operation: (acc: Int, Int) -> Int): Int {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun LongArray.reduce(operation: (acc: Long, Long) -> Long): Long {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun FloatArray.reduce(operation: (acc: Float, Float) -> Float): Float {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun DoubleArray.reduce(operation: (acc: Double, Double) -> Double): Double {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun BooleanArray.reduce(operation: (acc: Boolean, Boolean) -> Boolean): Boolean {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun CharArray.reduce(operation: (acc: Char, Char) -> Char): Char {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current

accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun <S, T : S> Array<out T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator: S = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun ByteArray.reduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): Byte {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun ShortArray.reduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): Short {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun IntArray.reduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): Int {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun LongArray.reduceIndexed(operation: (index: Int, acc: Long, Long) -> Long): Long {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an

element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun FloatArray.reduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): Float {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun DoubleArray.reduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): Double {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun BooleanArray.reduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): Boolean {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun CharArray.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <S, T : S> Array<out T>.reduceIndexedOrNull(operation: (index: Int, acc: S, T) -> S): S? {\n if (isEmpty())\n return null\n var accumulator: S = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun ByteArray.reduceIndexedOrNull(operation: (index: Int, acc: Byte, Byte) -> Byte): Byte? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to

right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun ShortArray.reduceIndexedOrNull(operation: (index: Int, acc: Short, Short) -> Short): Short? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun IntArray.reduceIndexedOrNull(operation: (index: Int, acc: Int, Int) -> Int): Int? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun LongArray.reduceIndexedOrNull(operation: (index: Int, acc: Long, Long) -> Long): Long? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun FloatArray.reduceIndexedOrNull(operation: (index: Int, acc: Float, Float) -> Float): Float? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun DoubleArray.reduceIndexedOrNull(operation: (index: Int, acc: Double, Double) -> Double): Double? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun BooleanArray.reduceIndexedOrNull(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): Boolean? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n

*/\n@SinceKotlin(\"1.4\")\npublic inline fun CharArray.reduceIndexedOrNull(operation: (index: Int, acc: Char, Char) -> Char): Char? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(index, accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <S, T : S> Array<out T>.reduceOrNull(operation: (acc: S, T) -> S): S? {\n   if (isEmpty())\n   return null\n   var accumulator: S = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun ByteArray.reduceOrNull(operation: (acc: Byte, Byte) -> Byte): Byte? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun ShortArray.reduceOrNull(operation: (acc: Short, Short) -> Short): Short? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun IntArray.reduceOrNull(operation: (acc: Int, Int) -> Int): Int? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun LongArray.reduceOrNull(operation: (acc: Long, Long) -> Long): Long? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun FloatArray.reduceOrNull(operation: (acc: Float, Float) -> Float): Float? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next

accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun DoubleArray.reduceOrNull(operation: (acc: Double, Double) -> Double): Double? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun BooleanArray.reduceOrNull(operation: (acc: Boolean, Boolean) -> Boolean): Boolean? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun CharArray.reduceOrNull(operation: (acc: Char, Char) -> Char): Char? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun <S, T : S> Array<out T>.reduceRight(operation: (T, acc: S) -> S): S {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator: S = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun ByteArray.reduceRight(operation: (Byte, acc: Byte) -> Byte): Byte {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun ShortArray.reduceRight(operation: (Short, acc: Short) -> Short): Short {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an

expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun IntArray.reduceRight(operation: (Int, acc: Int) -> Int): Int {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun LongArray.reduceRight(operation: (Long, acc: Long) -> Long): Long {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun FloatArray.reduceRight(operation: (Float, acc: Float) -> Float): Float {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun DoubleArray.reduceRight(operation: (Double, acc: Double) -> Double): Double {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun BooleanArray.reduceRight(operation: (Boolean, acc: Boolean) -> Boolean): Boolean {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun CharArray.reduceRight(operation: (Char, acc: Char) -> Char): Char {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while

(index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun <S, T : S> Array<out T>.reduceRightIndexed(operation: (index: Int, T, acc: S) -> S): S {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator: S = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun ByteArray.reduceRightIndexed(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun ShortArray.reduceRightIndexed(operation: (index: Int, Short, acc: Short) -> Short): Short {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun IntArray.reduceRightIndexed(operation: (index: Int, Int, acc: Int) -> Int): Int {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun LongArray.reduceRightIndexed(operation: (index: Int, Long, acc: Long) -> Long): Long {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index),

accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun FloatArray.reduceRightIndexed(operation: (index: Int, Float, acc: Float) -> Float): Float {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun DoubleArray.reduceRightIndexed(operation: (index: Int, Double, acc: Double) -> Double): Double {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun BooleanArray.reduceRightIndexed(operation: (index: Int, Boolean, acc: Boolean) -> Boolean): Boolean {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun CharArray.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <S, T : S> Array<out T>.reduceRightIndexedOrNull(operation: (index: Int, T, acc: S) -> S): S? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator: S = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n *

Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun ByteArray.reduceRightIndexedOrNull(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun ShortArray.reduceRightIndexedOrNull(operation: (index: Int, Short, acc: Short) -> Short): Short? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun IntArray.reduceRightIndexedOrNull(operation: (index: Int, Int, acc: Int) -> Int): Int? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun LongArray.reduceRightIndexedOrNull(operation: (index: Int, Long, acc: Long) -> Long): Long? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun FloatArray.reduceRightIndexedOrNull(operation: (index: Int, Float, acc: Float) -> Float): Float? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun DoubleArray.reduceRightIndexedOrNull(operation: (index: Int, Double, acc: Double) -> Double): Double? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n *

Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun BooleanArray.reduceRightIndexedOrNull(operation: (index: Int, Boolean, acc: Boolean) -> Boolean): Boolean? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun CharArray.reduceRightIndexedOrNull(operation: (index: Int, Char, acc: Char) -> Char): Char? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <S, T : S> Array<out T>.reduceRightOrNull(operation: (T, acc: S) -> S): S? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator: S = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun ByteArray.reduceRightOrNull(operation: (Byte, acc: Byte) -> Byte): Byte? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun ShortArray.reduceRightOrNull(operation: (Short, acc: Short) -> Short): Short? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun IntArray.reduceRightOrNull(operation: (Int, acc: Int) -> Int): Int? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and

applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun LongArray.reduceRightOrNull(operation: (Long, acc: Long) -> Long): Long? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun FloatArray.reduceRightOrNull(operation: (Float, acc: Float) -> Float): Float? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun DoubleArray.reduceRightOrNull(operation: (Double, acc: Double) -> Double): Double? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun BooleanArray.reduceRightOrNull(operation: (Boolean, acc: Boolean) -> Boolean): Boolean? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun CharArray.reduceRightOrNull(operation: (Char, acc: Char) -> Char): Char? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, R> Array<out T>.runningFold(initial: R, operation: (acc: R, T) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (element in this) {\n   accumulator = operation(accumulator, element)\n   result.add(accumulator)\n   }\n

return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> ByteArray.runningFold(initial: R, operation: (acc: R, Byte) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (element in this) {\n   accumulator = operation(accumulator, element)\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> ShortArray.runningFold(initial: R, operation: (acc: R, Short) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (element in this) {\n   accumulator = operation(accumulator, element)\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> IntArray.runningFold(initial: R, operation: (acc: R, Int) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (element in this) {\n   accumulator = operation(accumulator, element)\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> LongArray.runningFold(initial: R, operation: (acc: R, Long) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (element in this) {\n   accumulator = operation(accumulator, element)\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> FloatArray.runningFold(initial: R, operation: (acc: R, Float) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (element in this) {\n   accumulator = operation(accumulator, element)\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation]

function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> DoubleArray.runningFold(initial: R, operation: (acc: R, Double) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (element in this) {\n   accumulator = operation(accumulator, element)\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> BooleanArray.runningFold(initial: R, operation: (acc: R, Boolean) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (element in this) {\n   accumulator = operation(accumulator, element)\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharArray.runningFold(initial: R, operation: (acc: R, Char) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (element in this) {\n   accumulator = operation(accumulator, element)\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, R> Array<out T>.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (index in indices) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> ByteArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (index in indices) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation]

function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> ShortArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (index in indices) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> IntArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (index in indices) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> LongArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Long) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (index in indices) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> FloatArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (index in indices) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> DoubleArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (index in indices) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and

current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> BooleanArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (index in indices) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (index in indices) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and the element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <S, T : S> Array<out T>.runningReduce(operation: (acc: S, T) -> S): List<S> {\n if (isEmpty()) return emptyList()\n var accumulator: S = this[0]\n val result = ArrayList<S>(size).apply { add(accumulator) }\n for (index in 1 until size) {\n accumulator = operation(accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.runningReduce(operation: (acc: Byte, Byte) -> Byte): List<Byte> {\n if (isEmpty()) return emptyList()\n var accumulator = this[0]\n val result = ArrayList<Byte>(size).apply { add(accumulator) }\n for (index in 1 until size) {\n accumulator = operation(accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.runningReduce(operation: (acc: Short, Short) -> Short): List<Short> {\n if (isEmpty()) return emptyList()\n var accumulator = this[0]\n val result = ArrayList<Short>(size).apply { add(accumulator) }\n for (index in 1 until size) {\n accumulator = operation(accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n

*/\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.runningReduce(operation: (acc: Int, Int) -> Int): List<Int> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Int>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n
*/\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.runningReduce(operation: (acc: Long, Long) -> Long): List<Long> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Long>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n
*/\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.runningReduce(operation: (acc: Float, Float) -> Float): List<Float> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Float>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n
*/\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.runningReduce(operation: (acc: Double, Double) -> Double): List<Double> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Double>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n
*/\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.runningReduce(operation: (acc: Boolean, Boolean) -> Boolean): List<Boolean> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Boolean>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n
*/\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.runningReduce(operation: (acc: Char, Char) -> Char): List<Char> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Char>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current

accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <S, T : S> Array<out T>.runningReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {\n   if (isEmpty()) return emptyList()\n   var accumulator: S = this[0]\n   val result = ArrayList<S>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.runningReduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): List<Byte> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Byte>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.runningReduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): List<Short> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Short>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.runningReduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): List<Int> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Int>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.runningReduceIndexed(operation: (index: Int, acc: Long, Long) -> Long): List<Long> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Long>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.runningReduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): List<Float> {\n   if

(isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Float>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.runningReduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): List<Double> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Double>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.runningReduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): List<Boolean> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Boolean>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.runningReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Char>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <T, R> Array<out T>.scan(initial: R, operation: (acc: R, T) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> ByteArray.scan(initial: R, operation: (acc: R, Byte) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in

resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> ShortArray.scan(initial: R, operation: (acc: R, Short) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> IntArray.scan(initial: R, operation: (acc: R, Int) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> LongArray.scan(initial: R, operation: (acc: R, Long) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> FloatArray.scan(initial: R, operation: (acc: R, Float) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> DoubleArray.scan(initial: R, operation: (acc: R, Double) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> BooleanArray.scan(initial: R, operation: (acc: R, Boolean) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample

samples.collections.Collections.Aggregates.scan\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli
c inline fun <R> CharArray.scan(initial: R, operation: (acc: R, Char) -> R): List<R> {\n return runningFold(initial,
operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation]
from left to right\n * to each element, its index in the original array and current accumulator value that starts with
[initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it
would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an
element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n *
@sample samples.collections.Collections.Aggregates.scan\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <T, R>
Array<out T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {\n return
runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values
generated by applying [operation] from left to right\n * to each element, its index in the original array and current
accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should
not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation]
function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the
next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli
c inline fun <R> ByteArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): List<R> {\n return
runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values
generated by applying [operation] from left to right\n * to each element, its index in the original array and current
accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should
not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation]
function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the
next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli
c inline fun <R> ShortArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): List<R> {\n return
runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values
generated by applying [operation] from left to right\n * to each element, its index in the original array and current
accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should
not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation]
function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the
next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli
c inline fun <R> IntArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): List<R> {\n return
runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values
generated by applying [operation] from left to right\n * to each element, its index in the original array and current
accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should
not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation]
function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the
next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli
c inline fun <R> LongArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Long) -> R): List<R> {\n return
runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values
generated by applying [operation] from left to right\n * to each element, its index in the original array and current
accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should
not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation]
function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the

next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> FloatArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): List<R> {\n return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> DoubleArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): List<R> {\n return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> BooleanArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): List<R> {\n return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {\n return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun <T> Array<out T>.sumBy(selector: (T) -> Int): Int {\n var sum: Int = 0\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun ByteArray.sumBy(selector: (Byte) -> Int): Int {\n var sum: Int = 0\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun ShortArray.sumBy(selector: (Short) -> Int): Int {\n var sum: Int = 0\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun IntArray.sumBy(selector: (Int) -> Int): Int {\n var sum: Int = 0\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun LongArray.sumBy(selector: (Long) -> Int): Int {\n var sum: Int = 0\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function

applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
FloatArray.sumBy(selector: (Float) -> Int): Int {\n var sum: Int = 0\n for (element in this) {\n sum +=
selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function
applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
DoubleArray.sumBy(selector: (Double) -> Int): Int {\n var sum: Int = 0\n for (element in this) {\n sum +=
selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function
applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
BooleanArray.sumBy(selector: (Boolean) -> Int): Int {\n var sum: Int = 0\n for (element in this) {\n sum +=
selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function
applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
CharArray.sumBy(selector: (Char) -> Int): Int {\n var sum: Int = 0\n for (element in this) {\n sum +=
selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function
applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun <T>
Array<out T>.sumByDouble(selector: (T) -> Double): Double {\n var sum: Double = 0.0\n for (element in this)
{\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
ByteArray.sumByDouble(selector: (Byte) -> Double): Double {\n var sum: Double = 0.0\n for (element in this)
{\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
ShortArray.sumByDouble(selector: (Short) -> Double): Double {\n var sum: Double = 0.0\n for (element in this)
{\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
IntArray.sumByDouble(selector: (Int) -> Double): Double {\n var sum: Double = 0.0\n for (element in this) {\n
sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector]
function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
LongArray.sumByDouble(selector: (Long) -> Double): Double {\n var sum: Double = 0.0\n for (element in this)
{\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
FloatArray.sumByDouble(selector: (Float) -> Double): Double {\n var sum: Double = 0.0\n for (element in this)
{\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
DoubleArray.sumByDouble(selector: (Double) -> Double): Double {\n var sum: Double = 0.0\n for (element in
this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\",
ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun
BooleanArray.sumByDouble(selector: (Boolean) -> Double): Double {\n var sum: Double = 0.0\n for (element
in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced

by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun CharArray.sumByDouble(selector: (Char) -> Double): Double {\n var sum: Double = 0.0\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.sumOf(selector: (T) -> Double): Double {\n var sum: Double = 0.toDouble()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.sumOf(selector: (Byte) -> Double): Double {\n var sum: Double = 0.toDouble()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.sumOf(selector: (Short) -> Double): Double {\n var sum: Double = 0.toDouble()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.sumOf(selector: (Int) -> Double): Double {\n var sum: Double = 0.toDouble()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.sumOf(selector: (Long) -> Double): Double {\n var sum: Double = 0.toDouble()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.sumOf(selector: (Float) -> Double): Double {\n var sum: Double = 0.toDouble()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.sumOf(selector: (Double) -> Double): Double {\n var sum: Double = 0.toDouble()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.sumOf(selector: (Boolean) -> Double): Double {\n var sum: Double = 0.toDouble()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.sumOf(selector: (Char) -> Double): Double {\n var sum: Double = 0.toDouble()\n for (element in

this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.sumOf(selector: (T) -> Int): Int {\n var sum: Int = 0.toInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.sumOf(selector: (Byte) -> Int): Int {\n var sum: Int = 0.toInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.sumOf(selector: (Short) -> Int): Int {\n var sum: Int = 0.toInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.sumOf(selector: (Int) -> Int): Int {\n var sum: Int = 0.toInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.sumOf(selector: (Long) -> Int): Int {\n var sum: Int = 0.toInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.sumOf(selector: (Float) -> Int): Int {\n var sum: Int = 0.toInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.sumOf(selector: (Double) -> Int): Int {\n var sum: Int = 0.toInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.sumOf(selector: (Boolean) -> Int): Int {\n var sum: Int = 0.toInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.sumOf(selector: (Char) -> Int): Int {\n var sum: Int = 0.toInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.sumOf(selector: (T) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.sumOf(selector: (Byte) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.sumOf(selector: (Short) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.sumOf(selector: (Int) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.sumOf(selector: (Long) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.sumOf(selector: (Float) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.sumOf(selector: (Double) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.sumOf(selector: (Boolean) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.sumOf(selector: (Char) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.sumOf(selector: (T) -> UInt): UInt {\n var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return

sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.sumOf(selector: (Byte) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.sumOf(selector: (Short) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.sumOf(selector: (Int) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.sumOf(selector: (Long) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.sumOf(selector: (Float) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.sumOf(selector: (Double) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.sumOf(selector: (Boolean) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.sumOf(selector: (Char) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.sumOf(selector: (T) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n

return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.sumOf(selector: (Byte) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.sumOf(selector: (Short) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.sumOf(selector: (Int) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.sumOf(selector: (Long) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.sumOf(selector: (Float) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.sumOf(selector: (Double) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.sumOf(selector: (Boolean) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
*/\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.sumOf(selector: (Char) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return

```
sum\n}\n\n/**\n * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.\n */\npublic fun <T : Any> Array<T?>.requireNoNulls(): Array<T> {\n    for (element in this) {\n        if (element == null) {\n            throw IllegalArgumentException(\"null element found in $this.\")\n        }\n    }\n    @Suppress(\"UNCHECKED_CAST\")\n    return this as Array<T>\n}\n\n/**\n * Splits the original array into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n * @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives\n */\npublic inline fun <T> Array<out T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {\n    val first = ArrayList<T>()\n    val second = ArrayList<T>()\n    for (element in this) {\n        if (predicate(element)) {\n            first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n * @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives\n */\npublic inline fun ByteArray.partition(predicate: (Byte) -> Boolean): Pair<List<Byte>, List<Byte>> {\n    val first = ArrayList<Byte>()\n    val second = ArrayList<Byte>()\n    for (element in this) {\n        if (predicate(element)) {\n            first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n * @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives\n */\npublic inline fun ShortArray.partition(predicate: (Short) -> Boolean): Pair<List<Short>, List<Short>> {\n    val first = ArrayList<Short>()\n    val second = ArrayList<Short>()\n    for (element in this) {\n        if (predicate(element)) {\n            first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n * @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives\n */\npublic inline fun IntArray.partition(predicate: (Int) -> Boolean): Pair<List<Int>, List<Int>> {\n    val first = ArrayList<Int>()\n    val second = ArrayList<Int>()\n    for (element in this) {\n        if (predicate(element)) {\n            first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n * @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives\n */\npublic inline fun LongArray.partition(predicate: (Long) -> Boolean): Pair<List<Long>, List<Long>> {\n    val first = ArrayList<Long>()\n    val second = ArrayList<Long>()\n    for (element in this) {\n        if (predicate(element)) {\n            first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n * @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives\n */\npublic inline fun FloatArray.partition(predicate: (Float) -> Boolean): Pair<List<Float>, List<Float>> {\n    val first = ArrayList<Float>()\n    val second = ArrayList<Float>()\n    for (element in this) {\n        if (predicate(element)) {\n            first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n * @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives\n */\npublic inline fun DoubleArray.partition(predicate: (Double) -> Boolean): Pair<List<Double>, List<Double>> {\n    val first = ArrayList<Double>()\n    val second = ArrayList<Double>()\n    for (element in this) {\n        if (predicate(element)) {\n            first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair of lists,\n * where *first* list contains elements for which
```

[predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n * @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives\n */\npublic inline fun BooleanArray.partition(predicate: (Boolean) -> Boolean): Pair<List<Boolean>, List<Boolean>> {\n val first = ArrayList<Boolean>()\n val second = ArrayList<Boolean>()\n for (element in this) {\n if (predicate(element)) {\n first.add(element)\n } else {\n second.add(element)\n }\n }\n return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n * @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives\n */\npublic inline fun CharArray.partition(predicate: (Char) -> Boolean): Pair<List<Char>, List<Char>> {\n val first = ArrayList<Char>()\n val second = ArrayList<Char>()\n for (element in this) {\n if (predicate(element)) {\n first.add(element)\n } else {\n second.add(element)\n }\n }\n return Pair(first, second)\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <T, R> Array<out T>.zip(other: Array<out R>): List<Pair<T, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> ByteArray.zip(other: Array<out R>): List<Pair<Byte, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> ShortArray.zip(other: Array<out R>): List<Pair<Short, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> IntArray.zip(other: Array<out R>): List<Pair<Int, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> LongArray.zip(other: Array<out R>): List<Pair<Long, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> FloatArray.zip(other: Array<out R>): List<Pair<Float, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> DoubleArray.zip(other: Array<out R>): List<Pair<Double, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> BooleanArray.zip(other: Array<out R>): List<Pair<Boolean, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> CharArray.zip(other: Array<out R>): List<Pair<Char, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <T, R, V> Array<out T>.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return

list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> ByteArray.zip(other: Array<out R>, transform: (a: Byte, b: R) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> ShortArray.zip(other: Array<out R>, transform: (a: Short, b: R) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> IntArray.zip(other: Array<out R>, transform: (a: Int, b: R) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> LongArray.zip(other: Array<out R>, transform: (a: Long, b: R) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> FloatArray.zip(other: Array<out R>, transform: (a: Float, b: R) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> DoubleArray.zip(other: Array<out R>, transform: (a: Double, b: R) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> BooleanArray.zip(other: Array<out R>, transform: (a: Boolean, b: R) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> CharArray.zip(other: Array<out R>, transform: (a: Char, b: R) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample

samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <T, R> Array<out T>.zip(other: Iterable<R>): List<Pair<T, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> ByteArray.zip(other: Iterable<R>): List<Pair<Byte, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> ShortArray.zip(other: Iterable<R>): List<Pair<Short, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> IntArray.zip(other: Iterable<R>): List<Pair<Int, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> LongArray.zip(other: Iterable<R>): List<Pair<Long, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> FloatArray.zip(other: Iterable<R>): List<Pair<Float, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> DoubleArray.zip(other: Iterable<R>): List<Pair<Double, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> CharArray.zip(other: Iterable<R>): List<Pair<Char, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <T, R, V> Array<out T>.zip(other: Iterable<R>, transform: (a: T, b: R) -> V): List<V> {\n   val arraySize = size\n   val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n   var i = 0\n   for (element in other) {\n   if (i >= arraySize) break\n   list.add(transform(this[i++], element))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> ByteArray.zip(other: Iterable<R>, transform: (a: Byte, b: R) -> V): List<V> {\n   val arraySize = size\n   val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n   var i = 0\n   for (element in other) {\n   if (i >= arraySize) break\n   list.add(transform(this[i++], element))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> ShortArray.zip(other: Iterable<R>, transform: (a: Short, b: R) -> V): List<V> {\n   val arraySize = size\n   val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n   var i = 0\n   for (element in other) {\n   if (i >= arraySize) break\n   list.add(transform(this[i++], element))\n   }\n   return

list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> IntArray.zip(other: Iterable<R>, transform: (a: Int, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> LongArray.zip(other: Iterable<R>, transform: (a: Long, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> FloatArray.zip(other: Iterable<R>, transform: (a: Float, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> DoubleArray.zip(other: Iterable<R>, transform: (a: Double, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> BooleanArray.zip(other: Iterable<R>, transform: (a: Boolean, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> CharArray.zip(other: Iterable<R>, transform: (a: Char, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun ByteArray.zip(other: ByteArray): List<Pair<Byte, Byte>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun ShortArray.zip(other: ShortArray): List<Pair<Short, Short>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample

samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun IntArray.zip(other: IntArray): List<Pair<Int, Int>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun LongArray.zip(other: LongArray): List<Pair<Long, Long>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun FloatArray.zip(other: FloatArray): List<Pair<Float, Float>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun DoubleArray.zip(other: DoubleArray): List<Pair<Double, Double>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun BooleanArray.zip(other: BooleanArray): List<Pair<Boolean, Boolean>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun CharArray.zip(other: CharArray): List<Pair<Char, Char>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> ByteArray.zip(other: ByteArray, transform: (a: Byte, b: Byte) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> ShortArray.zip(other: ShortArray, transform: (a: Short, b: Short) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> IntArray.zip(other: IntArray, transform: (a: Int, b: Int) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> LongArray.zip(other: LongArray, transform: (a: Long, b: Long) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> FloatArray.zip(other: FloatArray, transform: (a: Float, b: Float) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array

with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> DoubleArray.zip(other: DoubleArray, transform: (a: Double, b: Double) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> CharArray.zip(other: CharArray, transform: (a: Char, b: Char) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <T, A : Appendable> Array<out T>.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((T) -> CharSequence)? = null): A {\n buffer.append(prefix)\n var count = 0\n for (element in this) {\n if (++count > 1) buffer.append(separator)\n if (limit < 0 || count <= limit) {\n buffer.appendElement(element, transform)\n } else break\n }\n if (limit >= 0 && count > limit) buffer.append(truncated)\n buffer.append(postfix)\n return buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A : Appendable> ByteArray.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Byte) -> CharSequence)? = null): A {\n buffer.append(prefix)\n var count = 0\n for (element in this) {\n if (++count > 1) buffer.append(separator)\n if (limit < 0 || count <= limit) {\n if (transform != null)\n buffer.append(transform(element))\n else\n buffer.append(element.toString())\n } else break\n }\n if (limit >= 0 && count > limit) buffer.append(truncated)\n buffer.append(postfix)\n return buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A : Appendable> ShortArray.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Short) -> CharSequence)? = null): A {\n buffer.append(prefix)\n var count = 0\n for (element in this) {\n if (++count > 1) buffer.append(separator)\n if (limit < 0 || count <= limit) {\n if (transform != null)\n buffer.append(transform(element))\n else\n buffer.append(element.toString())\n } else break\n }\n if (limit >= 0 && count > limit) buffer.append(truncated)\n buffer.append(postfix)\n return buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of

[limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A : Appendable> IntArray.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Int) -> CharSequence)? = null): A {\n buffer.append(prefix)\n var count = 0\n for (element in this) {\n if (++count > 1) buffer.append(separator)\n if (limit < 0 || count <= limit) {\n if (transform != null)\n buffer.append(transform(element))\n else\n buffer.append(element.toString())\n } else break\n }\n if (limit >= 0 && count > limit) buffer.append(truncated)\n buffer.append(postfix)\n return buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A : Appendable> LongArray.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Long) -> CharSequence)? = null): A {\n buffer.append(prefix)\n var count = 0\n for (element in this) {\n if (++count > 1) buffer.append(separator)\n if (limit < 0 || count <= limit) {\n if (transform != null)\n buffer.append(transform(element))\n else\n buffer.append(element.toString())\n } else break\n }\n if (limit >= 0 && count > limit) buffer.append(truncated)\n buffer.append(postfix)\n return buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A : Appendable> FloatArray.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Float) -> CharSequence)? = null): A {\n buffer.append(prefix)\n var count = 0\n for (element in this) {\n if (++count > 1) buffer.append(separator)\n if (limit < 0 || count <= limit) {\n if (transform != null)\n buffer.append(transform(element))\n else\n buffer.append(element.toString())\n } else break\n }\n if (limit >= 0 && count > limit) buffer.append(truncated)\n buffer.append(postfix)\n return buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A : Appendable> DoubleArray.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Double) -> CharSequence)? = null): A {\n buffer.append(prefix)\n var count = 0\n for (element in this) {\n if (++count > 1) buffer.append(separator)\n if (limit < 0 || count <= limit) {\n if (transform != null)\n buffer.append(transform(element))\n else\n buffer.append(element.toString())\n } else break\n }\n if (limit >= 0 && count > limit) buffer.append(truncated)\n buffer.append(postfix)\n return buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A : Appendable> BooleanArray.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Boolean) -> CharSequence)? = null): A {\n buffer.append(prefix)\n var count = 0\n for (element in this) {\n if (++count > 1) buffer.append(separator)\n if (limit < 0 || count <= limit) {\n if (transform != null)\n buffer.append(transform(element))\n else\n buffer.append(element.toString())\n } else break\n }\n if (limit >= 0 && count > limit) buffer.append(truncated)\n buffer.append(postfix)\n return

buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A : Appendable> CharArray.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Char) -> CharSequence)? = null): A {\n   buffer.append(prefix)\n   var count = 0\n   for (element in this) {\n   if (++count > 1) buffer.append(separator)\n   if (limit < 0 || count <= limit) {\n   if (transform != null)\n   buffer.append(transform(element))\n   else\n   buffer.append(element)\n   } else break\n   }\n   if (limit >= 0 && count > limit) buffer.append(truncated)\n   buffer.append(postfix)\n   return buffer\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun <T> Array<out T>.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((T) -> CharSequence)? = null): String {\n   return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun ByteArray.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Byte) -> CharSequence)? = null): String {\n   return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun ShortArray.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Short) -> CharSequence)? = null): String {\n   return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun IntArray.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Int) -> CharSequence)? = null): String {\n   return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun LongArray.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Long) -> CharSequence)? = null): String {\n   return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample

samples.collections.Collections.Transformations.joinToString\n */\npublic fun FloatArray.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Float) -> CharSequence)? = null): String {\n   return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun DoubleArray.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Double) -> CharSequence)? = null): String {\n   return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun BooleanArray.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Boolean) -> CharSequence)? = null): String {\n   return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun CharArray.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((Char) -> CharSequence)? = null): String {\n   return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n */\npublic fun <T> Array<out T>.asIterable(): Iterable<T> {\n   if (isEmpty()) return emptyList()\n   return Iterable { this.iterator() }\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n */\npublic fun ByteArray.asIterable(): Iterable<Byte> {\n   if (isEmpty()) return emptyList()\n   return Iterable { this.iterator() }\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n */\npublic fun ShortArray.asIterable(): Iterable<Short> {\n   if (isEmpty()) return emptyList()\n   return Iterable { this.iterator() }\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n */\npublic fun IntArray.asIterable(): Iterable<Int> {\n   if (isEmpty()) return emptyList()\n   return Iterable { this.iterator() }\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n */\npublic fun LongArray.asIterable(): Iterable<Long> {\n   if (isEmpty()) return emptyList()\n   return Iterable { this.iterator() }\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n */\npublic fun FloatArray.asIterable(): Iterable<Float> {\n   if (isEmpty()) return emptyList()\n   return Iterable { this.iterator() }\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n */\npublic fun DoubleArray.asIterable(): Iterable<Double> {\n   if (isEmpty()) return emptyList()\n   return Iterable { this.iterator() }\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n */\npublic fun BooleanArray.asIterable(): Iterable<Boolean> {\n   if (isEmpty()) return emptyList()\n   return Iterable { this.iterator() }\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n */\npublic fun CharArray.asIterable(): Iterable<Char> {\n   if (isEmpty()) return emptyList()\n   return Iterable { this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromArray\n */\npublic fun <T> Array<out T>.asSequence(): Sequence<T> {\n   if (isEmpty()) return emptySequence()\n   return Sequence { this.iterator() }\n}\n\n/**\n *

Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromArray\n */\npublic fun ByteArray.asSequence(): Sequence<Byte> {\n   if (isEmpty()) return emptySequence()\n   return Sequence { this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromArray\n */\npublic fun ShortArray.asSequence(): Sequence<Short> {\n   if (isEmpty()) return emptySequence()\n   return Sequence { this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromArray\n */\npublic fun IntArray.asSequence(): Sequence<Int> {\n   if (isEmpty()) return emptySequence()\n   return Sequence { this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromArray\n */\npublic fun LongArray.asSequence(): Sequence<Long> {\n   if (isEmpty()) return emptySequence()\n   return Sequence { this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromArray\n */\npublic fun FloatArray.asSequence(): Sequence<Float> {\n   if (isEmpty()) return emptySequence()\n   return Sequence { this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromArray\n */\npublic fun DoubleArray.asSequence(): Sequence<Double> {\n   if (isEmpty()) return emptySequence()\n   return Sequence { this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromArray\n */\npublic fun BooleanArray.asSequence(): Sequence<Boolean> {\n   if (isEmpty()) return emptySequence()\n   return Sequence { this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromArray\n */\npublic fun CharArray.asSequence(): Sequence<Char> {\n   if (isEmpty()) return emptySequence()\n   return Sequence { this.iterator() }\n}\n\n/**\n * Returns an average value of elements in the array.\n */\n@kotlin.jvm.JvmName(\"averageOfByte\")\npublic fun Array<out Byte>.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n */\n@kotlin.jvm.JvmName(\"averageOfShort\")\npublic fun Array<out Short>.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n */\n@kotlin.jvm.JvmName(\"averageOfInt\")\npublic fun Array<out Int>.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n */\n@kotlin.jvm.JvmName(\"averageOfLong\")\npublic fun Array<out Long>.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n */\n@kotlin.jvm.JvmName(\"averageOfFloat\")\npublic fun Array<out Float>.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n */\n@kotlin.jvm.JvmName(\"averageOfDouble\")\npublic fun Array<out Double>.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n */\npublic fun ByteArray.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n */\npublic fun ShortArray.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of

elements in the array.\n */\npublic fun IntArray.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n */\npublic fun LongArray.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n */\npublic fun FloatArray.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n */\npublic fun DoubleArray.average(): Double {\n   var sum: Double = 0.0\n   var count: Int = 0\n   for (element in this) {\n   sum += element\n   ++count\n   }\n   return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName(\"sumOfByte\")\npublic fun Array<out Byte>.sum(): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName(\"sumOfShort\")\npublic fun Array<out Short>.sum(): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName(\"sumOfInt\")\npublic fun Array<out Int>.sum(): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName(\"sumOfLong\")\npublic fun Array<out Long>.sum(): Long {\n   var sum: Long = 0L\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName(\"sumOfFloat\")\npublic fun Array<out Float>.sum(): Float {\n   var sum: Float = 0.0f\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName(\"sumOfDouble\")\npublic fun Array<out Double>.sum(): Double {\n   var sum: Double = 0.0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\npublic fun ByteArray.sum(): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\npublic fun ShortArray.sum(): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\npublic fun IntArray.sum(): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\npublic fun LongArray.sum(): Long {\n   var sum: Long = 0L\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\npublic fun FloatArray.sum(): Float {\n   var sum: Float = 0.0f\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\npublic fun DoubleArray.sum(): Double {\n   var sum: Double = 0.0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"RangesKt\")\n\npackage kotlin.ranges\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\n\n/**\n * Returns a random element from this range.\n * \n * @throws IllegalArgumentException if this range is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun IntRange.random(): Int {\n   return random(Random)\n}\n\n/**\n * Returns a random element from this range.\n * \n * @throws IllegalArgumentException if this range is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun LongRange.random(): Long {\n   return random(Random)\n}\n\n/**\n * Returns a random element from this range.\n * \n * @throws IllegalArgumentException if this range is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun CharRange.random(): Char {\n   return random(Random)\n}\n\n/**\n * Returns a random element from this range using the specified source of randomness.\n * \n * @throws IllegalArgumentException if this range is empty.\n

*/\n@SinceKotlin(\"1.3\")\npublic fun IntRange.random(random: Random): Int {\n try {\n return random.nextInt(this)\n } catch(e: IllegalArgumentException) {\n throw NoSuchElementException(e.message)\n }\n}\n\n/**\n * Returns a random element from this range using the specified source of randomness.\n * \n * @throws IllegalArgumentException if this range is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun LongRange.random(random: Random): Long {\n try {\n return random.nextLong(this)\n } catch(e: IllegalArgumentException) {\n throw NoSuchElementException(e.message)\n }\n}\n\n/**\n * Returns a random element from this range using the specified source of randomness.\n * \n * @throws IllegalArgumentException if this range is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun CharRange.random(random: Random): Char {\n try {\n return random.nextInt(first.code, last.code + 1).toChar()\n } catch(e: IllegalArgumentException) {\n throw NoSuchElementException(e.message)\n }\n}\n\n/**\n * Returns a random element from this range, or `null` if this range is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun IntRange.randomOrNull(): Int? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this range, or `null` if this range is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun LongRange.randomOrNull(): Long? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this range, or `null` if this range is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun CharRange.randomOrNull(): Char? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun IntRange.randomOrNull(random: Random): Int? {\n if (isEmpty())\n return null\n return random.nextInt(this)\n}\n\n/**\n * Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun LongRange.randomOrNull(random: Random): Long? {\n if (isEmpty())\n return null\n return random.nextLong(this)\n}\n\n/**\n * Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun CharRange.randomOrNull(random: Random): Char? {\n if (isEmpty())\n return null\n return random.nextInt(first.code, last.code + 1).toChar()\n}\n\n/**\n * Returns `true` if this range contains the specified [element].\n * \n * Always returns `false` if the [element] is `null`.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline operator fun IntRange.contains(element: Int?): Boolean {\n return element != null && contains(element)\n}\n\n/**\n * Returns `true` if this range contains the specified [element].\n * \n * Always returns `false` if the [element] is `null`.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline operator fun LongRange.contains(element: Long?): Boolean {\n return element != null && contains(element)\n}\n\n/**\n * Returns `true` if this range contains the specified [element].\n * \n * Always returns `false` if the [element] is `null`.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline operator fun CharRange.contains(element: Char?): Boolean {\n return element != null && contains(element)\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"intRangeContains\")\npublic operator fun ClosedRange<Int>.contains(value: Byte): Boolean {\n return contains(value.toInt())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"longRangeContains\")\npublic operator fun ClosedRange<Long>.contains(value: Byte): Boolean {\n return contains(value.toLong())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"shortRangeContains\")\npublic operator fun ClosedRange<Short>.contains(value: Byte): Boolean {\n return contains(value.toShort())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer

and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"doubleRangeContains\")\npublic operator fun ClosedRange<Double>.contains(value: Byte): Boolean {\n   return contains(value.toDouble())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"floatRangeContains\")\npublic operator fun ClosedRange<Float>.contains(value: Byte): Boolean {\n   return contains(value.toFloat())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"intRangeContains\")\npublic operator fun ClosedRange<Int>.contains(value: Double): Boolean {\n   return value.toIntExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"longRangeContains\")\npublic operator fun ClosedRange<Long>.contains(value: Double): Boolean {\n   return value.toLongExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"byteRangeContains\")\npublic operator fun ClosedRange<Byte>.contains(value: Double): Boolean {\n   return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"shortRangeContains\")\npublic operator fun ClosedRange<Short>.contains(value: Double): Boolean {\n   return value.toShortExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"floatRangeContains\")\npublic operator fun ClosedRange<Float>.contains(value: Double): Boolean {\n   return contains(value.toFloat())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"intRangeContains\")\npublic operator fun ClosedRange<Int>.contains(value: Float): Boolean {\n   return value.toIntExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"longRangeContains\")\npublic operator fun ClosedRange<Long>.contains(value: Float): Boolean {\n   return value.toLongExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"byteRangeContains\")\npublic operator fun ClosedRange<Byte>.contains(value: Float): Boolean {\n   return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince

= \"1.5\")\n@kotlin.jvm.JvmName(\"shortRangeContains\")\npublic operator fun ClosedRange<Short>.contains(value: Float): Boolean {\n   return value.toShortExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"doubleRangeContains\")\npublic operator fun ClosedRange<Double>.contains(value: Float): Boolean {\n   return contains(value.toDouble())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"longRangeContains\")\npublic operator fun ClosedRange<Long>.contains(value: Int): Boolean {\n   return contains(value.toLong())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"byteRangeContains\")\npublic operator fun ClosedRange<Byte>.contains(value: Int): Boolean {\n   return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"shortRangeContains\")\npublic operator fun ClosedRange<Short>.contains(value: Int): Boolean {\n   return value.toShortExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"doubleRangeContains\")\npublic operator fun ClosedRange<Double>.contains(value: Int): Boolean {\n   return contains(value.toDouble())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"floatRangeContains\")\npublic operator fun ClosedRange<Float>.contains(value: Int): Boolean {\n   return contains(value.toFloat())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"intRangeContains\")\npublic operator fun ClosedRange<Int>.contains(value: Long): Boolean {\n   return value.toIntExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"byteRangeContains\")\npublic operator fun ClosedRange<Byte>.contains(value: Long): Boolean {\n   return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"shortRangeContains\")\npublic operator fun ClosedRange<Short>.contains(value: Long): Boolean {\n   return value.toShortExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"doubleRangeContains\")\npublic operator fun ClosedRange<Double>.contains(value: Long): Boolean {\n   return contains(value.toDouble())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"floatRangeContains\")\npublic operator fun ClosedRange<Float>.contains(value: Long): Boolean {\n   return contains(value.toFloat())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"intRangeContains\")\npublic operator fun ClosedRange<Int>.contains(value: Short): Boolean {\n   return contains(value.toInt())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"longRangeContains\")\npublic operator fun ClosedRange<Long>.contains(value: Short): Boolean {\n   return contains(value.toLong())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@kotlin.jvm.JvmName(\"byteRangeContains\")\npublic operator fun ClosedRange<Byte>.contains(value: Short): Boolean {\n   return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince

= \"1.5\")\n@kotlin.jvm.JvmName(\"doubleRangeContains\")\npublic operator fun ClosedRange<Double>.contains(value: Short): Boolean {\n   return contains(value.toDouble())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@Deprecated(\"This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = \"1.3\", errorSince = \"1.4\", hiddenSince = \"1.5\")\n@kotlin.jvm.JvmName(\"floatRangeContains\")\npublic operator fun ClosedRange<Float>.contains(value: Short): Boolean {\n   return contains(value.toFloat())\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Int.downTo(to: Byte): IntProgression {\n   return IntProgression.fromClosedRange(this, to.toInt(), -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Long.downTo(to: Byte): LongProgression {\n   return LongProgression.fromClosedRange(this, to.toLong(), -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Byte.downTo(to: Byte): IntProgression {\n   return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Short.downTo(to: Byte): IntProgression {\n   return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Char.downTo(to: Char): CharProgression {\n   return CharProgression.fromClosedRange(this, to, -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Int.downTo(to: Int): IntProgression {\n   return IntProgression.fromClosedRange(this, to, -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Long.downTo(to: Int): LongProgression {\n   return LongProgression.fromClosedRange(this, to.toLong(), -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Byte.downTo(to: Int): IntProgression {\n   return IntProgression.fromClosedRange(this.toInt(), to, -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Short.downTo(to: Int): IntProgression {\n   return IntProgression.fromClosedRange(this.toInt(), to, -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Int.downTo(to: Long): LongProgression {\n   return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Long.downTo(to: Long): LongProgression {\n   return LongProgression.fromClosedRange(this, to, -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Byte.downTo(to: Long): LongProgression {\n   return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less

than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Short.downTo(to: Long): LongProgression {\n return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Int.downTo(to: Short): IntProgression {\n return IntProgression.fromClosedRange(this, to.toInt(), -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Long.downTo(to: Short): LongProgression {\n return LongProgression.fromClosedRange(this, to.toLong(), -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Byte.downTo(to: Short): IntProgression {\n return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Short.downTo(to: Short): IntProgression {\n return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n}\n\n/**\n * Returns a progression that goes over the same range in the opposite direction with the same step.\n */\npublic fun IntProgression.reversed(): IntProgression {\n return IntProgression.fromClosedRange(last, first, -step)\n}\n\n/**\n * Returns a progression that goes over the same range in the opposite direction with the same step.\n */\npublic fun LongProgression.reversed(): LongProgression {\n return LongProgression.fromClosedRange(last, first, -step)\n}\n\n/**\n * Returns a progression that goes over the same range in the opposite direction with the same step.\n */\npublic fun CharProgression.reversed(): CharProgression {\n return CharProgression.fromClosedRange(last, first, -step)\n}\n\n/**\n * Returns a progression that goes over the same range with the given step.\n */\npublic infix fun IntProgression.step(step: Int): IntProgression {\n checkStepIsPositive(step > 0, step)\n return IntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}\n\n/**\n * Returns a progression that goes over the same range with the given step.\n */\npublic infix fun LongProgression.step(step: Long): LongProgression {\n checkStepIsPositive(step > 0, step)\n return LongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}\n\n/**\n * Returns a progression that goes over the same range with the given step.\n */\npublic infix fun CharProgression.step(step: Int): CharProgression {\n checkStepIsPositive(step > 0, step)\n return CharProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}\n\ninternal fun Int.toByteExactOrNull(): Byte? {\n return if (this in Byte.MIN_VALUE.toInt()..Byte.MAX_VALUE.toInt()) this.toByte() else null\n}\n\ninternal fun Long.toByteExactOrNull(): Byte? {\n return if (this in Byte.MIN_VALUE.toLong()..Byte.MAX_VALUE.toLong()) this.toByte() else null\n}\n\ninternal fun Short.toByteExactOrNull(): Byte? {\n return if (this in Byte.MIN_VALUE.toShort()..Byte.MAX_VALUE.toShort()) this.toByte() else null\n}\n\ninternal fun Double.toByteExactOrNull(): Byte? {\n return if (this in Byte.MIN_VALUE.toDouble()..Byte.MAX_VALUE.toDouble()) this.toInt().toByte() else null\n}\n\ninternal fun Float.toByteExactOrNull(): Byte? {\n return if (this in Byte.MIN_VALUE.toFloat()..Byte.MAX_VALUE.toFloat()) this.toInt().toByte() else null\n}\n\ninternal fun Long.toIntExactOrNull(): Int? {\n return if (this in Int.MIN_VALUE.toLong()..Int.MAX_VALUE.toLong()) this.toInt() else null\n}\n\ninternal fun Double.toIntExactOrNull(): Int? {\n return if (this in Int.MIN_VALUE.toDouble()..Int.MAX_VALUE.toDouble()) this.toInt() else null\n}\n\ninternal fun Float.toIntExactOrNull(): Int? {\n return if (this in Int.MIN_VALUE.toFloat()..Int.MAX_VALUE.toFloat()) this.toInt() else null\n}\n\ninternal fun Double.toLongExactOrNull(): Long? {\n return if (this in Long.MIN_VALUE.toDouble()..Long.MAX_VALUE.toDouble()) this.toLong() else null\n}\n\ninternal fun Float.toLongExactOrNull(): Long? {\n return if (this in Long.MIN_VALUE.toFloat()..Long.MAX_VALUE.toFloat()) this.toLong() else null\n}\n\ninternal fun

Int.toShortExactOrNull(): Short? {\n   return if (this in Short.MIN_VALUE.toInt()..Short.MAX_VALUE.toInt()) this.toShort() else null\n}\n\ninternal fun Long.toShortExactOrNull(): Short? {\n   return if (this in Short.MIN_VALUE.toLong()..Short.MAX_VALUE.toLong()) this.toShort() else null\n}\n\ninternal fun Double.toShortExactOrNull(): Short? {\n   return if (this in Short.MIN_VALUE.toDouble()..Short.MAX_VALUE.toDouble()) this.toInt().toShort() else null\n}\n\ninternal fun Float.toShortExactOrNull(): Short? {\n   return if (this in Short.MIN_VALUE.toFloat()..Short.MAX_VALUE.toFloat()) this.toInt().toShort() else null\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Int.until(to: Byte): IntRange {\n   return this .. (to.toInt() - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Long.until(to: Byte): LongRange {\n   return this .. (to.toLong() - 1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Byte.until(to: Byte): IntRange {\n   return this.toInt() .. (to.toInt() - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Short.until(to: Byte): IntRange {\n   return this.toInt() .. (to.toInt() - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Char.until(to: Char): CharRange {\n   if (to <= '\\u0000') return CharRange.EMPTY\n   return this .. (to - 1).toChar()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Int.until(to: Int): IntRange {\n   if (to <= Int.MIN_VALUE) return IntRange.EMPTY\n   return this .. (to - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Long.until(to: Int): LongRange {\n   return this .. (to.toLong() - 1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Byte.until(to: Int): IntRange {\n   if (to <= Int.MIN_VALUE) return IntRange.EMPTY\n   return this.toInt() .. (to - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Short.until(to: Int): IntRange {\n   if (to <= Int.MIN_VALUE) return IntRange.EMPTY\n   return this.toInt() .. (to - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Int.until(to: Long): LongRange {\n   if (to <= Long.MIN_VALUE) return LongRange.EMPTY\n   return this.toLong() .. (to - 1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Long.until(to: Long): LongRange {\n   if (to <= Long.MIN_VALUE) return LongRange.EMPTY\n   return this .. (to - 1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Byte.until(to: Long): LongRange {\n   if (to <= Long.MIN_VALUE) return LongRange.EMPTY\n   return this.toLong() .. (to - 1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Short.until(to: Long): LongRange {\n   if (to <= Long.MIN_VALUE) return LongRange.EMPTY\n   return this.toLong() .. (to - 1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Int.until(to: Short): IntRange {\n   return this .. (to.toInt() - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Long.until(to: Short): LongRange {\n   return this .. (to.toLong() -

1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Byte.until(to: Short): IntRange {\n   return this.toInt() .. (to.toInt() - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Short.until(to: Short): IntRange {\n   return this.toInt() .. (to.toInt() - 1).toInt()\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeastComparable\n */\npublic fun <T : Comparable<T>> T.coerceAtLeast(minimumValue: T): T {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Byte.coerceAtLeast(minimumValue: Byte): Byte {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Short.coerceAtLeast(minimumValue: Short): Short {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Int.coerceAtLeast(minimumValue: Int): Int {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Long.coerceAtLeast(minimumValue: Long): Long {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Float.coerceAtLeast(minimumValue: Float): Float {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Double.coerceAtLeast(minimumValue: Double): Double {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMostComparable\n */\npublic fun <T : Comparable<T>> T.coerceAtMost(maximumValue: T): T {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun Byte.coerceAtMost(maximumValue: Byte): Byte {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun Short.coerceAtMost(maximumValue: Short): Short {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun Int.coerceAtMost(maximumValue: Int): Int {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or

the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun Long.coerceAtMost(maximumValue: Long): Long {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun Float.coerceAtMost(maximumValue: Float): Float {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun Double.coerceAtMost(maximumValue: Double): Double {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceInComparable\n */\npublic fun <T : Comparable<T>> T.coerceIn(minimumValue: T?, maximumValue: T?): T {\n   if (minimumValue !== null && maximumValue !== null) {\n   if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n   if (this < minimumValue) return minimumValue\n   if (this > maximumValue) return maximumValue\n   }\n   else {\n   if (minimumValue !== null && this < minimumValue) return minimumValue\n   if (maximumValue !== null && this > maximumValue) return maximumValue\n   }\n   return this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Byte.coerceIn(minimumValue: Byte, maximumValue: Byte): Byte {\n   if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n   if (this < minimumValue) return minimumValue\n   if (this > maximumValue) return maximumValue\n   return this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Short.coerceIn(minimumValue: Short, maximumValue: Short): Short {\n   if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n   if (this < minimumValue) return minimumValue\n   if (this > maximumValue) return maximumValue\n   return this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Int.coerceIn(minimumValue: Int, maximumValue: Int): Int {\n   if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n   if (this < minimumValue) return minimumValue\n   if (this > maximumValue) return maximumValue\n   return this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Long.coerceIn(minimumValue: Long, maximumValue: Long): Long {\n   if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n   if (this < minimumValue) return minimumValue\n   if (this > maximumValue) return maximumValue\n   return this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value

is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Float.coerceIn(minimumValue: Float, maximumValue: Float): Float {\n if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n if (this < minimumValue) return minimumValue\n if (this > maximumValue) return maximumValue\n return this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Double.coerceIn(minimumValue: Double, maximumValue: Double): Double {\n if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n if (this < minimumValue) return minimumValue\n if (this > maximumValue) return maximumValue\n return this\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * \n * @return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.\n * \n * @sample samples.comparisons.ComparableOps.coerceInFloatingPointRange\n */\n@SinceKotlin(\"1.1\")\npublic fun <T : Comparable<T>> T.coerceIn(range: ClosedFloatingPointRange<T>): T {\n if (range.isEmpty()) throw IllegalArgumentException(\"Cannot coerce value to an empty range: $range.\")\n return when {\n // this < start equiv to this <= start && !(this >= start)\n range.lessThanOrEquals(this, range.start) && !range.lessThanOrEquals(range.start, this) -> range.start\n // this > end equiv to this >= end && !(this <= end)\n range.lessThanOrEquals(range.endInclusive, this) && !range.lessThanOrEquals(this, range.endInclusive) -> range.endInclusive\n else -> this\n }\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * \n * @return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.\n * \n * @sample samples.comparisons.ComparableOps.coerceInComparable\n */\npublic fun <T : Comparable<T>> T.coerceIn(range: ClosedRange<T>): T {\n if (range is ClosedFloatingPointRange) {\n return this.coerceIn<T>(range)\n }\n if (range.isEmpty()) throw IllegalArgumentException(\"Cannot coerce value to an empty range: $range.\")\n return when {\n this < range.start -> range.start\n this > range.endInclusive -> range.endInclusive\n else -> this\n }\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * \n * @return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.\n * \n * @sample samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Int.coerceIn(range: ClosedRange<Int>): Int {\n if (range is ClosedFloatingPointRange) {\n return this.coerceIn<Int>(range)\n }\n if (range.isEmpty()) throw IllegalArgumentException(\"Cannot coerce value to an empty range: $range.\")\n return when {\n this < range.start -> range.start\n this > range.endInclusive -> range.endInclusive\n else -> this\n }\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * \n * @return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.\n * \n * @sample samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Long.coerceIn(range: ClosedRange<Long>): Long {\n if (range is ClosedFloatingPointRange) {\n return this.coerceIn<Long>(range)\n }\n if (range.isEmpty()) throw IllegalArgumentException(\"Cannot coerce value to an empty range: $range.\")\n return when {\n this < range.start -> range.start\n this > range.endInclusive -> range.endInclusive\n else -> this\n }\n}\n\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin\n\nimport kotlin.experimental.*\nimport kotlin.jvm.*\n\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@JvmInline\npublic value class UByte @PublishedApi internal constructor(@PublishedApi internal val data: Byte) : Comparable<UByte> {\n\n companion object {\n /**\n * A constant holding the minimum value an instance of UByte can have.\n */\n public const val MIN_VALUE: UByte = UByte(0)\n\n /**\n *

A constant holding the maximum value an instance of UByte can have.\n */\n public const val MAX_VALUE: UByte = UByte(-1)\n\n /**\n * The number of bytes used to represent an instance of UByte in a binary form.\n */\n public const val SIZE_BYTES: Int = 1\n\n /**\n * The number of bits used to represent an instance of UByte in a binary form.\n */\n public const val SIZE_BITS: Int = 8\n }\n\n /**\n * Compares this value with the specified value for order.\n * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n * or a positive number if it's greater than other.\n */\n @kotlin.internal.InlineOnly\n @Suppress(\"OVERRIDE_BY_INLINE\")\n public override inline operator fun compareTo(other: UByte): Int = this.toInt().compareTo(other.toInt())\n\n /**\n * Compares this value with the specified value for order.\n * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n * or a positive number if it's greater than other.\n */\n @kotlin.internal.InlineOnly\n public inline operator fun compareTo(other: UShort): Int = this.toInt().compareTo(other.toInt())\n\n /**\n * Compares this value with the specified value for order.\n * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n * or a positive number if it's greater than other.\n */\n @kotlin.internal.InlineOnly\n public inline operator fun compareTo(other: UInt): Int = this.toUInt().compareTo(other)\n\n /**\n * Compares this value with the specified value for order.\n * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n * or a positive number if it's greater than other.\n */\n @kotlin.internal.InlineOnly\n public inline operator fun compareTo(other: ULong): Int = this.toULong().compareTo(other)\n\n /** Adds the other value to this value. */\n @kotlin.internal.InlineOnly\n public inline operator fun plus(other: UByte): UInt = this.toUInt().plus(other.toUInt())\n /** Adds the other value to this value. */\n @kotlin.internal.InlineOnly\n public inline operator fun plus(other: UShort): UInt = this.toUInt().plus(other.toUInt())\n /** Adds the other value to this value. */\n @kotlin.internal.InlineOnly\n public inline operator fun plus(other: UInt): UInt = this.toUInt().plus(other)\n /** Adds the other value to this value. */\n @kotlin.internal.InlineOnly\n public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)\n\n /** Subtracts the other value from this value. */\n @kotlin.internal.InlineOnly\n public inline operator fun minus(other: UByte): UInt = this.toUInt().minus(other.toUInt())\n /** Subtracts the other value from this value. */\n @kotlin.internal.InlineOnly\n public inline operator fun minus(other: UShort): UInt = this.toUInt().minus(other.toUInt())\n /** Subtracts the other value from this value. */\n @kotlin.internal.InlineOnly\n public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other)\n /** Subtracts the other value from this value. */\n @kotlin.internal.InlineOnly\n public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)\n\n /** Multiplies this value by the other value. */\n @kotlin.internal.InlineOnly\n public inline operator fun times(other: UByte): UInt = this.toUInt().times(other.toUInt())\n /** Multiplies this value by the other value. */\n @kotlin.internal.InlineOnly\n public inline operator fun times(other: UShort): UInt = this.toUInt().times(other.toUInt())\n /** Multiplies this value by the other value. */\n @kotlin.internal.InlineOnly\n public inline operator fun times(other: UInt): UInt = this.toUInt().times(other)\n /** Multiplies this value by the other value. */\n @kotlin.internal.InlineOnly\n public inline operator fun times(other: ULong): ULong = this.toULong().times(other)\n\n /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n @kotlin.internal.InlineOnly\n public inline operator fun div(other: UByte): UInt = this.toUInt().div(other.toUInt())\n /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n @kotlin.internal.InlineOnly\n public inline operator fun div(other: UShort): UInt = this.toUInt().div(other.toUInt())\n /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n @kotlin.internal.InlineOnly\n public inline operator fun div(other: UInt): UInt = this.toUInt().div(other)\n /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n @kotlin.internal.InlineOnly\n public inline operator fun div(other: ULong): ULong = this.toULong().div(other)\n\n /**\n * Calculates the remainder of truncating division of this value by the other value.\n * \n * The result is always less than the divisor.\n */\n @kotlin.internal.InlineOnly\n public inline operator fun rem(other: UByte): UInt =

this.toUInt().rem(other.toUInt())\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt())\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other)\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)\n\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UByte): UInt = this.toUInt().floorDiv(other.toUInt())\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UShort): UInt = this.toUInt().floorDiv(other.toUInt())\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UInt): UInt = this.toUInt().floorDiv(other)\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: ULong): ULong = this.toULong().floorDiv(other)\n\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: UByte): UByte = this.toUInt().mod(other.toUInt()).toUByte()\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: UShort): UShort = this.toUInt().mod(other.toUInt()).toUShort()\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: UInt): UInt = this.toUInt().mod(other)\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: ULong): ULong = this.toULong().mod(other)\n\n   /**\n   * Returns this value incremented by one.\n   *\n   * @sample samples.misc.Builtins.inc\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun inc(): UByte = UByte(data.inc())\n\n   /**\n   * Returns this value decremented by one.\n   *\n   * @sample samples.misc.Builtins.dec\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun dec(): UByte = UByte(data.dec())\n\n   /** Creates a range from this value to the specified [other] value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rangeTo(other: UByte): UIntRange = UIntRange(this.toUInt(), other.toUInt())\n\n   /** Performs a bitwise AND operation between the two values. */\n   @kotlin.internal.InlineOnly\n   public inline infix fun and(other: UByte): UByte = UByte(this.data and other.data)\n   /** Performs a bitwise OR operation between the two values. */\n   @kotlin.internal.InlineOnly\n   public inline infix fun or(other: UByte): UByte = UByte(this.data or other.data)\n   /** Performs a bitwise XOR operation between the two values. */\n   @kotlin.internal.InlineOnly\n   public inline infix fun xor(other: UByte): UByte = UByte(this.data xor other.data)\n   /** Inverts the bits in this value. */\n   @kotlin.internal.InlineOnly\n   public inline fun inv(): UByte = UByte(data.inv())\n\n   /**\n   * Converts this [UByte] value to [Byte].\n   *\n   * If this value is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents\n   * the same numerical value as this `UByte`. Otherwise the result is negative.\n   *\n   * The resulting `Byte` value has the

same binary representation as this `UByte` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toByte(): Byte = data\n /**\n * Converts this [UByte] value to [Short].\n *\n * The resulting `Short` value represents the same numerical value as this `UByte`.\n *\n * The least significant 8 bits of the resulting `Short` value are the same as the bits of this `UByte` value,\n * whereas the most significant 8 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toShort(): Short = data.toShort() and 0xFF\n /**\n * Converts this [UByte] value to [Int].\n *\n * The resulting `Int` value represents the same numerical value as this `UByte`.\n *\n * The least significant 8 bits of the resulting `Int` value are the same as the bits of this `UByte` value,\n * whereas the most significant 24 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toInt(): Int = data.toInt() and 0xFF\n /**\n * Converts this [UByte] value to [Long].\n *\n * The resulting `Long` value represents the same numerical value as this `UByte`.\n *\n * The least significant 8 bits of the resulting `Long` value are the same as the bits of this `UByte` value,\n * whereas the most significant 56 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toLong(): Long = data.toLong() and 0xFF\n\n /** Returns this value. */\n @kotlin.internal.InlineOnly\n public inline fun toUByte(): UByte = this\n /**\n * Converts this [UByte] value to [UShort].\n *\n * The resulting `UShort` value represents the same numerical value as this `UByte`.\n *\n * The least significant 8 bits of the resulting `UShort` value are the same as the bits of this `UByte` value,\n * whereas the most significant 8 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUShort(): UShort = UShort(data.toShort() and 0xFF)\n /**\n * Converts this [UByte] value to [UInt].\n *\n * The resulting `UInt` value represents the same numerical value as this `UByte`.\n *\n * The least significant 8 bits of the resulting `UInt` value are the same as the bits of this `UByte` value,\n * whereas the most significant 24 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUInt(): UInt = UInt(data.toInt() and 0xFF)\n /**\n * Converts this [UByte] value to [ULong].\n *\n * The resulting `ULong` value represents the same numerical value as this `UByte`.\n *\n * The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte` value,\n * whereas the most significant 56 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toULong(): ULong = ULong(data.toLong() and 0xFF)\n\n /**\n * Converts this [UByte] value to [Float].\n *\n * The resulting `Float` value represents the same numerical value as this `UByte`.\n */\n @kotlin.internal.InlineOnly\n public inline fun toFloat(): Float = this.toInt().toFloat()\n /**\n * Converts this [UByte] value to [Double].\n *\n * The resulting `Double` value represents the same numerical value as this `UByte`.\n */\n @kotlin.internal.InlineOnly\n public inline fun toDouble(): Double = this.toInt().toDouble()\n\n public override fun toString(): String = toInt().toString()\n\n}\n\n/**\n * Converts this [Byte] value to [UByte].\n *\n * If this value is positive, the resulting `UByte` value represents the same numerical value as this `Byte`.\n *\n * The resulting `UByte` value has the same binary representation as this `Byte` value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Byte.toUByte(): UByte = UByte(this)\n/**\n * Converts this [Short] value to [UByte].\n *\n * If this value is positive and less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents\n * the same numerical value as this `Short`.\n *\n * The resulting `UByte` value is represented by the least significant 8 bits of this `Short` value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Short.toUByte(): UByte = UByte(this.toByte())\n/**\n * Converts this [Int] value to [UByte].\n *\n * If this value is positive and less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents\n * the same numerical value as this `Int`.\n *\n * The resulting `UByte` value is represented by the least significant 8 bits of this `Int` value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Int.toUByte(): UByte = UByte(this.toByte())\n/**\n * Converts this [Long] value to [UByte].\n *\n * If this value is positive and less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents\n * the same numerical value as this `Long`.\n *\n * The resulting `UByte` value is represented by the

least significant 8 bits of this `Long` value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Long.toUByte(): UByte = UByte(this.toByte())\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin\n\nimport kotlin.experimental.*\nimport kotlin.jvm.*\n\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@JvmInline\npublic value class UInt @PublishedApi internal constructor(@PublishedApi internal val data: Int) : Comparable<UInt> {\n\n   companion object {\n   /**\n   * A constant holding the minimum value an instance of UInt can have.\n   */\n   public const val MIN_VALUE: UInt = UInt(0)\n\n   /**\n   * A constant holding the maximum value an instance of UInt can have.\n   */\n   public const val MAX_VALUE: UInt = UInt(-1)\n\n   /**\n   * The number of bytes used to represent an instance of UInt in a binary form.\n   */\n   public const val SIZE_BYTES: Int = 4\n\n   /**\n   * The number of bits used to represent an instance of UInt in a binary form.\n   */\n   public const val SIZE_BITS: Int = 32\n   }\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   @Suppress(\"OVERRIDE_BY_INLINE\")\n   public override inline operator fun compareTo(other: UInt): Int = uintCompare(this.data, other.data)\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun compareTo(other: ULong): Int = this.toULong().compareTo(other)\n\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: UByte): UInt = this.plus(other.toUInt())\n\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: UShort): UInt = this.plus(other.toUInt())\n\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: UInt): UInt = UInt(this.data.plus(other.data))\n\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)\n\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: UByte): UInt = this.minus(other.toUInt())\n\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: UShort): UInt = this.minus(other.toUInt())\n\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: UInt): UInt = UInt(this.data.minus(other.data))\n\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)\n\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: UByte): UInt = this.times(other.toUInt())\n\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: UShort): UInt = this.times(other.toUInt())\n\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: UInt): UInt = UInt(this.data.times(other.data))\n\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: ULong): ULong = this.toULong().times(other)\n\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: UByte): UInt =

this.div(other.toUInt())\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: UShort): UInt = this.div(other.toUInt())\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: UInt): UInt = uintDivide(this, other)\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: ULong): ULong = this.toULong().div(other)\n\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UByte): UInt = this.rem(other.toUInt())\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UShort): UInt = this.rem(other.toUInt())\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UInt): UInt = uintRemainder(this, other)\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)\n\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UByte): UInt = this.floorDiv(other.toUInt())\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UShort): UInt = this.floorDiv(other.toUInt())\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UInt): UInt = div(other)\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: ULong): ULong = this.toULong().floorDiv(other)\n\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: UByte): UByte = this.mod(other.toUInt()).toUByte()\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: UShort): UShort = this.mod(other.toUInt()).toUShort()\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: UInt): UInt = rem(other)\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: ULong): ULong = this.toULong().mod(other)\n\n   /**\n   * Returns this value incremented by one.\n   *\n   * @sample samples.misc.Builtins.inc\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun inc(): UInt = UInt(data.inc())\n\n   /**\n   * Returns this value decremented by one.\n   *\n   * @sample samples.misc.Builtins.dec\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun dec(): UInt = UInt(data.dec())\n\n   /** Creates a range from this value to the specified [other] value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rangeTo(other: UInt): UIntRange = UIntRange(this, other)\n\n   /**\n   * Shifts this value left by the [bitCount] number of bits.\n   *\n   * Note that only the five lowest-order bits of the [bitCount] are used as the shift distance.\n   * The shift distance actually used is therefore

always in the range `0..31`.\n */\n @kotlin.internal.InlineOnly\n public inline infix fun shl(bitCount: Int): UInt = UInt(data shl bitCount)\n\n /**\n * Shifts this value right by the [bitCount] number of bits, filling the leftmost bits with zeros.\n *\n * Note that only the five lowest-order bits of the [bitCount] are used as the shift distance.\n * The shift distance actually used is therefore always in the range `0..31`.\n */\n @kotlin.internal.InlineOnly\n public inline infix fun shr(bitCount: Int): UInt = UInt(data ushr bitCount)\n\n /** Performs a bitwise AND operation between the two values. */\n @kotlin.internal.InlineOnly\n public inline infix fun and(other: UInt): UInt = UInt(this.data and other.data)\n /** Performs a bitwise OR operation between the two values. */\n @kotlin.internal.InlineOnly\n public inline infix fun or(other: UInt): UInt = UInt(this.data or other.data)\n /** Performs a bitwise XOR operation between the two values. */\n @kotlin.internal.InlineOnly\n public inline infix fun xor(other: UInt): UInt = UInt(this.data xor other.data)\n /** Inverts the bits in this value. */\n @kotlin.internal.InlineOnly\n public inline fun inv(): UInt = UInt(data.inv())\n\n /**\n * Converts this [UInt] value to [Byte].\n *\n * If this value is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents\n * the same numerical value as this `UInt`.\n *\n * The resulting `Byte` value is represented by the least significant 8 bits of this `UInt` value.\n * Note that the resulting `Byte` value may be negative.\n */\n @kotlin.internal.InlineOnly\n public inline fun toByte(): Byte = data.toByte()\n /**\n * Converts this [UInt] value to [Short].\n *\n * If this value is less than or equals to [Short.MAX_VALUE], the resulting `Short` value represents\n * the same numerical value as this `UInt`.\n *\n * The resulting `Short` value is represented by the least significant 16 bits of this `UInt` value.\n * Note that the resulting `Short` value may be negative.\n */\n @kotlin.internal.InlineOnly\n public inline fun toShort(): Short = data.toShort()\n /**\n * Converts this [UInt] value to [Int].\n *\n * If this value is less than or equals to [Int.MAX_VALUE], the resulting `Int` value represents\n * the same numerical value as this `UInt`. Otherwise the result is negative.\n *\n * The resulting `Int` value has the same binary representation as this `UInt` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toInt(): Int = data\n /**\n * Converts this [UInt] value to [Long].\n *\n * The resulting `Long` value represents the same numerical value as this `UInt`.\n *\n * The least significant 32 bits of the resulting `Long` value are the same as the bits of this `UInt` value,\n * whereas the most significant 32 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toLong(): Long = data.toLong() and 0xFFFF_FFFF\n\n /**\n * Converts this [UInt] value to [UByte].\n *\n * If this value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents\n * the same numerical value as this `UInt`.\n *\n * The resulting `UByte` value is represented by the least significant 8 bits of this `UInt` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUByte(): UByte = data.toUByte()\n /**\n * Converts this [UInt] value to [UShort].\n *\n * If this value is less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents\n * the same numerical value as this `UInt`.\n *\n * The resulting `UShort` value is represented by the least significant 16 bits of this `UInt` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUShort(): UShort = data.toUShort()\n /** Returns this value. */\n @kotlin.internal.InlineOnly\n public inline fun toUInt(): UInt = this\n /**\n * Converts this [UInt] value to [ULong].\n *\n * The resulting `ULong` value represents the same numerical value as this `UInt`.\n *\n * The least significant 32 bits of the resulting `ULong` value are the same as the bits of this `UInt` value,\n * whereas the most significant 32 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toULong(): ULong = ULong(data.toLong() and 0xFFFF_FFFF)\n\n /**\n * Converts this [UInt] value to [Float].\n *\n * The resulting value is the closest `Float` to this `UInt` value.\n * In case when this `UInt` value is exactly between two `Float`s,\n * the one with zero at least significant bit of mantissa is selected.\n */\n @kotlin.internal.InlineOnly\n public inline fun toFloat(): Float = this.toDouble().toFloat()\n /**\n * Converts this [UInt] value to [Double].\n *\n * The resulting `Double` value represents the same numerical value as this `UInt`.\n */\n @kotlin.internal.InlineOnly\n public inline fun toDouble(): Double = uintToDouble(data)\n\n public override fun toString(): String = toLong().toString()\n\n}\n\n/**\n * Converts this [Byte] value to [UInt].\n *\n * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Byte`.\n *\n * The least significant 8 bits of the resulting `UInt` value are the same as the bits of this `Byte` value,\n * whereas the most significant 24 bits are filled with the sign bit of this value.\n

*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Byte.toUInt(): UInt = UInt(this.toInt())\n/**\n * Converts this [Short] value to [UInt].\n *\n * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Short`.\n *\n * The least significant 16 bits of the resulting `UInt` value are the same as the bits of this `Short` value,\n * whereas the most significant 16 bits are filled with the sign bit of this value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Short.toUInt(): UInt = UInt(this.toInt())\n/**\n * Converts this [Int] value to [UInt].\n *\n * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Int`.\n *\n * The resulting `UInt` value has the same binary representation as this `Int` value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Int.toUInt(): UInt = UInt(this)\n/**\n * Converts this [Long] value to [UInt].\n *\n * If this value is positive and less than or equals to [UInt.MAX_VALUE], the resulting `UInt` value represents\n * the same numerical value as this `Long`.\n *\n * The resulting `UInt` value is represented by the least significant 32 bits of this `Long` value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Long.toUInt(): UInt = UInt(this.toInt())\n\n/**\n * Converts this [Float] value to [UInt].\n *\n * The fractional part, if any, is rounded down towards zero.\n * Returns zero if this `Float` value is negative or `NaN`, [UInt.MAX_VALUE] if it's bigger than `UInt.MAX_VALUE`.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Float.toUInt(): UInt = doubleToUInt(this.toDouble())\n/**\n * Converts this [Double] value to [UInt].\n *\n * The fractional part, if any, is rounded down towards zero.\n * Returns zero if this `Double` value is negative or `NaN`, [UInt.MAX_VALUE] if it's bigger than `UInt.MAX_VALUE`.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Double.toUInt(): UInt = doubleToUInt(this)\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin\n\nimport kotlin.experimental.*\nimport kotlin.jvm.*\n\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@JvmInline\npublic value class UShort @PublishedApi internal constructor(@PublishedApi internal val data: Short) : Comparable<UShort> {\n\n   companion object {\n   /**\n   * A constant holding the minimum value an instance of UShort can have.\n   */\n   public const val MIN_VALUE: UShort = UShort(0)\n\n   /**\n   * A constant holding the maximum value an instance of UShort can have.\n   */\n   public const val MAX_VALUE: UShort = UShort(-1)\n\n   /**\n   * The number of bytes used to represent an instance of UShort in a binary form.\n   */\n   public const val SIZE_BYTES: Int = 2\n\n   /**\n   * The number of bits used to represent an instance of UShort in a binary form.\n   */\n   public const val SIZE_BITS: Int = 16\n   }\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun compareTo(other: UByte): Int = this.toInt().compareTo(other.toInt())\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   @Suppress(\"OVERRIDE_BY_INLINE\")\n   public override inline operator fun compareTo(other: UShort): Int = this.toInt().compareTo(other.toInt())\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun compareTo(other: UInt): Int = this.toUInt().compareTo(other)\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n

public inline operator fun compareTo(other: ULong): Int = this.toULong().compareTo(other)\n\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: UByte): UInt = this.toUInt().plus(other.toUInt())\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: UShort): UInt = this.toUInt().plus(other.toUInt())\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: UInt): UInt = this.toUInt().plus(other)\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)\n\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: UByte): UInt = this.toUInt().minus(other.toUInt())\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: UShort): UInt = this.toUInt().minus(other.toUInt())\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other)\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)\n\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: UByte): UInt = this.toUInt().times(other.toUInt())\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: UShort): UInt = this.toUInt().times(other.toUInt())\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: UInt): UInt = this.toUInt().times(other)\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: ULong): ULong = this.toULong().times(other)\n\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: UByte): UInt = this.toUInt().div(other.toUInt())\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: UShort): UInt = this.toUInt().div(other.toUInt())\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: UInt): UInt = this.toUInt().div(other)\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: ULong): ULong = this.toULong().div(other)\n\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UByte): UInt = this.toUInt().rem(other.toUInt())\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt())\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other)\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)\n\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UByte): UInt = this.toUInt().floorDiv(other.toUInt())\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UShort): UInt = this.toUInt().floorDiv(other.toUInt())\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UInt): UInt = this.toUInt().floorDiv(other)\n   /**\n   * Divides this value by the

other value, flooring the result to an integer that is closer to negative infinity.\n * \n * For unsigned types, the results of flooring division and truncating division are the same.\n */\n @kotlin.internal.InlineOnly\n public inline fun floorDiv(other: ULong): ULong = this.toULong().floorDiv(other)\n\n /**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is always less than the divisor.\n * \n * For unsigned types, the remainders of flooring division and truncating division are the same.\n */\n @kotlin.internal.InlineOnly\n public inline fun mod(other: UByte): UByte = this.toUInt().mod(other.toUInt()).toUByte()\n /**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is always less than the divisor.\n * \n * For unsigned types, the remainders of flooring division and truncating division are the same.\n */\n @kotlin.internal.InlineOnly\n public inline fun mod(other: UShort): UShort = this.toUInt().mod(other.toUInt()).toUShort()\n /**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is always less than the divisor.\n * \n * For unsigned types, the remainders of flooring division and truncating division are the same.\n */\n @kotlin.internal.InlineOnly\n public inline fun mod(other: UInt): UInt = this.toUInt().mod(other)\n /**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is always less than the divisor.\n * \n * For unsigned types, the remainders of flooring division and truncating division are the same.\n */\n @kotlin.internal.InlineOnly\n public inline fun mod(other: ULong): ULong = this.toULong().mod(other)\n\n /**\n * Returns this value incremented by one.\n *\n * @sample samples.misc.Builtins.inc\n */\n @kotlin.internal.InlineOnly\n public inline operator fun inc(): UShort = UShort(data.inc())\n\n /**\n * Returns this value decremented by one.\n *\n * @sample samples.misc.Builtins.dec\n */\n @kotlin.internal.InlineOnly\n public inline operator fun dec(): UShort = UShort(data.dec())\n\n /** Creates a range from this value to the specified [other] value. */\n @kotlin.internal.InlineOnly\n public inline operator fun rangeTo(other: UShort): UIntRange = UIntRange(this.toUInt(), other.toUInt())\n\n /** Performs a bitwise AND operation between the two values. */\n @kotlin.internal.InlineOnly\n public inline infix fun and(other: UShort): UShort = UShort(this.data and other.data)\n /** Performs a bitwise OR operation between the two values. */\n @kotlin.internal.InlineOnly\n public inline infix fun or(other: UShort): UShort = UShort(this.data or other.data)\n /** Performs a bitwise XOR operation between the two values. */\n @kotlin.internal.InlineOnly\n public inline infix fun xor(other: UShort): UShort = UShort(this.data xor other.data)\n /** Inverts the bits in this value. */\n @kotlin.internal.InlineOnly\n public inline fun inv(): UShort = UShort(data.inv())\n\n /**\n * Converts this [UShort] value to [Byte].\n *\n * If this value is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents\n * the same numerical value as this `UShort`.\n *\n * The resulting `Byte` value is represented by the least significant 8 bits of this `UShort` value.\n * Note that the resulting `Byte` value may be negative.\n */\n @kotlin.internal.InlineOnly\n public inline fun toByte(): Byte = data.toByte()\n /**\n * Converts this [UShort] value to [Short].\n *\n * If this value is less than or equals to [Short.MAX_VALUE], the resulting `Short` value represents\n * the same numerical value as this `UShort`. Otherwise the result is negative.\n *\n * The resulting `Short` value has the same binary representation as this `UShort` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toShort(): Short = data\n /**\n * Converts this [UShort] value to [Int].\n *\n * The resulting `Int` value represents the same numerical value as this `UShort`.\n *\n * The least significant 16 bits of the resulting `Int` value are the same as the bits of this `UShort` value,\n * whereas the most significant 16 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toInt(): Int = data.toInt() and 0xFFFF\n /**\n * Converts this [UShort] value to [Long].\n *\n * The resulting `Long` value represents the same numerical value as this `UShort`.\n *\n * The least significant 16 bits of the resulting `Long` value are the same as the bits of this `UShort` value,\n * whereas the most significant 48 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toLong(): Long = data.toLong() and 0xFFFF\n\n /**\n * Converts this [UShort] value to [UByte].\n *\n * If this value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents\n * the same numerical value as this `UShort`.\n *\n * The resulting `UByte` value is represented by the least significant 8 bits of this `UShort` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUByte(): UByte = data.toUByte()\n /** Returns this value. */\n

@kotlin.internal.InlineOnly\n public inline fun toUShort(): UShort = this\n /**\n * Converts this [UShort] value to [UInt].\n *\n * The resulting `UInt` value represents the same numerical value as this `UShort`.\n *\n * The least significant 16 bits of the resulting `UInt` value are the same as the bits of this `UShort` value,\n * whereas the most significant 16 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUInt(): UInt = UInt(data.toInt() and 0xFFFF)\n /**\n * Converts this [UShort] value to [ULong].\n *\n * The resulting `ULong` value represents the same numerical value as this `UShort`.\n *\n * The least significant 16 bits of the resulting `ULong` value are the same as the bits of this `UShort` value,\n * whereas the most significant 48 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toULong(): ULong = ULong(data.toLong() and 0xFFFF)\n\n /**\n * Converts this [UShort] value to [Float].\n *\n * The resulting `Float` value represents the same numerical value as this `UShort`.\n */\n @kotlin.internal.InlineOnly\n public inline fun toFloat(): Float = this.toInt().toFloat()\n /**\n * Converts this [UShort] value to [Double].\n *\n * The resulting `Double` value represents the same numerical value as this `UShort`.\n */\n @kotlin.internal.InlineOnly\n public inline fun toDouble(): Double = this.toInt().toDouble()\n\n public override fun toString(): String = toInt().toString()\n\n}\n\n/**\n * Converts this [Byte] value to [UShort].\n *\n * If this value is positive, the resulting `UShort` value represents the same numerical value as this `Byte`.\n *\n * The least significant 8 bits of the resulting `UShort` value are the same as the bits of this `Byte` value,\n * whereas the most significant 8 bits are filled with the sign bit of this value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Byte.toUShort(): UShort = UShort(this.toShort())\n/**\n * Converts this [Short] value to [UShort].\n *\n * If this value is positive, the resulting `UShort` value represents the same numerical value as this `Short`.\n *\n * The resulting `UShort` value has the same binary representation as this `Short` value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Short.toUShort(): UShort = UShort(this)\n/**\n * Converts this [Int] value to [UShort].\n *\n * If this value is positive and less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents\n * the same numerical value as this `Int`.\n *\n * The resulting `UShort` value is represented by the least significant 16 bits of this `Int` value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Int.toUShort(): UShort = UShort(this.toShort())\n/**\n * Converts this [Long] value to [UShort].\n *\n * If this value is positive and less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents\n * the same numerical value as this `Long`.\n *\n * The resulting `UShort` value is represented by the least significant 16 bits of this `Long` value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Long.toUShort(): UShort = UShort(this.toShort())\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin.ranges\n\n/**\n * A range of values of type `Char`.\n */\npublic class CharRange(start: Char, endInclusive: Char) : CharProgression(start, endInclusive, 1), ClosedRange<Char> {\n override val start: Char get() = first\n override val endInclusive: Char get() = last\n\n override fun contains(value: Char): Boolean = first <= value && value <= last\n\n /** \n * Checks whether the range is empty.\n *\n * The range is empty if its start value is greater than the end value.\n */\n override fun isEmpty(): Boolean = first > last\n\n override fun equals(other: Any?): Boolean =\n other is CharRange && (isEmpty() && other.isEmpty() ||\n first == other.first && last == other.last)\n\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * first.code + last.code)\n\n override fun toString(): String = \"$first..$last\"\n\n companion object {\n /** An empty range of values of type Char. */\n public val EMPTY: CharRange = CharRange(1.toChar(), 0.toChar())\n }\n}\n\n/**\n * A range of values of type `Int`.\n */\npublic class IntRange(start: Int, endInclusive: Int) : IntProgression(start, endInclusive, 1), ClosedRange<Int> {\n override val start: Int get() = first\n override val endInclusive: Int get() = last\n\n override fun contains(value: Int): Boolean = first <= value && value <= last\n\n /** \n * Checks whether the range is empty.\n *\n * The range is empty if its start value is greater than the end value.\n */\n

override fun isEmpty(): Boolean = first > last\n\n   override fun equals(other: Any?): Boolean =\n   other is IntRange && (isEmpty() && other.isEmpty() ||\n   first == other.first && last == other.last)\n\n   override fun hashCode(): Int =\n   if (isEmpty()) -1 else (31 * first + last)\n\n   override fun toString(): String = \"$first..$last\"\n\n   companion object {\n   /** An empty range of values of type Int. */\n   public val EMPTY: IntRange = IntRange(1, 0)\n   }\n}\n\n/**\n * A range of values of type `Long`.\n */\npublic class LongRange(start: Long, endInclusive: Long) : LongProgression(start, endInclusive, 1), ClosedRange<Long> {\n override val start: Long get() = first\n   override val endInclusive: Long get() = last\n\n   override fun contains(value: Long): Boolean = first <= value && value <= last\n\n   /** \n   * Checks whether the range is empty.\n   *\n   * The range is empty if its start value is greater than the end value.\n   */\n   override fun isEmpty(): Boolean = first > last\n\n   override fun equals(other: Any?): Boolean =\n   other is LongRange && (isEmpty() && other.isEmpty() ||\n   first == other.first && last == other.last)\n\n   override fun hashCode(): Int =\n   if (isEmpty()) -1 else (31 * (first xor (first ushr 32)) + (last xor (last ushr 32))).toInt()\n\n   override fun toString(): String = \"$first..$last\"\n\n   companion object {\n   /** An empty range of values of type Long. */\n   public val EMPTY: LongRange = LongRange(1, 0)\n   }\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"CollectionsKt\")\n@file:OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n\npackage kotlin.collections\n\nimport kotlin.contracts.*\nimport kotlin.random.Random\n\ninternal object EmptyIterator : ListIterator<Nothing> {\n   override fun hasNext(): Boolean = false\n   override fun hasPrevious(): Boolean = false\n   override fun nextIndex(): Int = 0\n   override fun previousIndex(): Int = -1\n   override fun next(): Nothing = throw NoSuchElementException()\n   override fun previous(): Nothing = throw NoSuchElementException()\n}\n\ninternal object EmptyList : List<Nothing>, Serializable, RandomAccess {\n   private const val serialVersionUID: Long = -7390468764508069838L\n\n override fun equals(other: Any?): Boolean = other is List<*> && other.isEmpty()\n   override fun hashCode(): Int = 1\n   override fun toString(): String = \"[]\"\n\n   override val size: Int get() = 0\n   override fun isEmpty(): Boolean = true\n   override fun contains(element: Nothing): Boolean = false\n   override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()\n\n   override fun get(index: Int): Nothing = throw IndexOutOfBoundsException(\"Empty list doesn't contain element at index $index.\")\n   override fun indexOf(element: Nothing): Int = -1\n   override fun lastIndexOf(element: Nothing): Int = -1\n\n   override fun iterator(): Iterator<Nothing> = EmptyIterator\n   override fun listIterator(): ListIterator<Nothing> = EmptyIterator\n   override fun listIterator(index: Int): ListIterator<Nothing> {\n   if (index != 0) throw IndexOutOfBoundsException(\"Index: $index\")\n   return EmptyIterator\n   }\n\n   override fun subList(fromIndex: Int, toIndex: Int): List<Nothing> {\n   if (fromIndex == 0 && toIndex == 0) return this\n   throw IndexOutOfBoundsException(\"fromIndex: $fromIndex, toIndex: $toIndex\")\n   }\n\n   private fun readResolve(): Any = EmptyList\n}\n\ninternal fun <T> Array<out T>.asCollection(): Collection<T> = ArrayAsCollection(this, isVarargs = false)\n\nprivate class ArrayAsCollection<T>(val values: Array<out T>, val isVarargs: Boolean) : Collection<T> {\n   override val size: Int get() = values.size\n   override fun isEmpty(): Boolean = values.isEmpty()\n   override fun contains(element: T): Boolean = values.contains(element)\n   override fun containsAll(elements: Collection<T>): Boolean = elements.all { contains(it) }\n   override fun iterator(): Iterator<T> = values.iterator()\n   // override hidden toArray implementation to prevent copying of values array\n   public fun toArray(): Array<out Any?> = values.copyToArrayOfAny(isVarargs)\n}\n\n/**\n * Returns an empty read-only list. The returned list is serializable (JVM).\n * @sample samples.collections.Collections.Lists.emptyReadOnlyList\n */\npublic fun <T> emptyList(): List<T> = EmptyList\n\n/**\n * Returns a new read-only list of given elements. The returned list is serializable (JVM).\n * @sample samples.collections.Collections.Lists.readOnlyList\n */\npublic fun <T> listOf(vararg elements: T): List<T> = if (elements.size > 0) elements.asList() else emptyList()\n\n/**\n * Returns an empty read-only list. The returned list is serializable (JVM).\n * @sample samples.collections.Collections.Lists.emptyReadOnlyList\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> listOf(): List<T> = emptyList()\n\n/**\n * Returns an empty

new [MutableList].\n * @sample samples.collections.Collections.Lists.emptyMutableList\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> mutableListOf(): MutableList<T> = ArrayList()\n\n/**\n * Returns an empty new [ArrayList].\n * @sample samples.collections.Collections.Lists.emptyArrayList\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> arrayListOf(): ArrayList<T> = ArrayList()\n\n/**\n * Returns a new [MutableList] with the given elements.\n * @sample samples.collections.Collections.Lists.mutableList\n */\npublic fun <T> mutableListOf(vararg elements: T): MutableList<T> =\n   if (elements.size == 0) ArrayList() else ArrayList(ArrayAsCollection(elements, isVarargs = true))\n\n/**\n * Returns a new [ArrayList] with the given elements.\n * @sample samples.collections.Collections.Lists.arrayList\n */\npublic fun <T> arrayListOf(vararg elements: T): ArrayList<T> =\n   if (elements.size == 0) ArrayList() else ArrayList(ArrayAsCollection(elements, isVarargs = true))\n\n/**\n * Returns a new read-only list either of single given element, if it is not null, or empty list if the element is null. The returned list is serializable (JVM).\n * @sample samples.collections.Collections.Lists.listOfNotNull\n */\npublic fun <T : Any> listOfNotNull(element: T?): List<T> = if (element != null) listOf(element) else emptyList()\n\n/**\n * Returns a new read-only list only of those given elements, that are not null. The returned list is serializable (JVM).\n * @sample samples.collections.Collections.Lists.listOfNotNull\n */\npublic fun <T : Any> listOfNotNull(vararg elements: T?): List<T> = elements.filterNotNull()\n\n/**\n * Creates a new read-only list with the specified [size], where each element is calculated by calling the specified\n * [init] function.\n *\n * The function [init] is called for each list element sequentially starting from the first one.\n * It should return the value for a list element given its index.\n *\n * @sample samples.collections.Collections.Lists.readOnlyListFromInitializer\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> List(size: Int, init: (index: Int) -> T): List<T> = MutableList(size, init)\n\n/**\n * Creates a new mutable list with the specified [size], where each element is calculated by calling the specified\n * [init] function.\n *\n * The function [init] is called for each list element sequentially starting from the first one.\n * It should return the value for a list element given its index.\n *\n * @sample samples.collections.Collections.Lists.mutableListFromInitializer\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> MutableList(size: Int, init: (index: Int) -> T): MutableList<T> {\n   val list = ArrayList<T>(size)\n   repeat(size) { index -> list.add(init(index)) }\n   return list\n}\n\n/**\n * Builds a new read-only [List] by populating a [MutableList] using the given [builderAction]\n * and returning a read-only list with the same elements.\n *\n * The list passed as a receiver to the [builderAction] is valid only inside that function.\n * Using it outside of the function produces an unspecified behavior.\n *\n * The returned list is serializable (JVM).\n *\n * @sample samples.collections.Builders.Lists.buildListSample\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <E> buildList(@BuilderInference builderAction: MutableList<E>.() -> Unit): List<E> {\n   contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }\n   return buildListInternal(builderAction)\n}\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal expect inline fun <E> buildListInternal(builderAction: MutableList<E>.() -> Unit): List<E>\n\n/**\n * Builds a new read-only [List] by populating a [MutableList] using the given [builderAction]\n * and returning a read-only list with the same elements.\n *\n * The list passed as a receiver to the [builderAction] is valid only inside that function.\n * Using it outside of the function produces an unspecified behavior.\n *\n * The returned list is serializable (JVM).\n *\n * [capacity] is used to hint the expected number of elements added in the [builderAction].\n *\n * @throws IllegalArgumentException if the given [capacity] is negative.\n *\n * @sample samples.collections.Builders.Lists.buildListSampleWithCapacity\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <E> buildList(capacity: Int, @BuilderInference builderAction: MutableList<E>.() -> Unit): List<E> {\n   contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }\n   return buildListInternal(capacity, builderAction)\n}\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal expect inline fun <E> buildListInternal(capacity: Int, builderAction: MutableList<E>.() -> Unit): List<E>\n\n/**\n * Returns an

[IntRange] of the valid indices for this collection.\n * @sample samples.collections.Collections.Collections.indicesOfCollection\n */\npublic val Collection<*>.indices: IntRange\n get() = 0..size - 1\n\n/**\n * Returns the index of the last item in the list or -1 if the list is empty.\n *\n * @sample samples.collections.Collections.Lists.lastIndexOfList\n */\npublic val <T> List<T>.lastIndex: Int\n get() = this.size - 1\n\n/**\n * Returns `true` if the collection is not empty.\n * @sample samples.collections.Collections.Collections.collectionIsNotEmpty\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Collection<T>.isNotEmpty(): Boolean = !isEmpty()\n\n/**\n * Returns `true` if this nullable collection is either null or empty.\n * @sample samples.collections.Collections.Collections.collectionIsNullOrEmpty\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Collection<T>?.isNullOrEmpty(): Boolean {\n contract {\n returns(false) implies (this@isNullOrEmpty != null)\n }\n\n return this == null || this.isEmpty()\n}\n\n/**\n * Returns this Collection if it's not `null` and the empty list otherwise.\n * @sample samples.collections.Collections.Collections.collectionOrEmpty\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Collection<T>?.orEmpty(): Collection<T> = this ?: emptyList()\n\n/**\n * Returns this List if it's not `null` and the empty list otherwise.\n * @sample samples.collections.Collections.Lists.listOrEmpty\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> List<T>?.orEmpty(): List<T> = this ?: emptyList()\n\n/**\n * Returns this collection if it's not empty\n * or the result of calling [defaultValue] function if the collection is empty.\n *\n * @sample samples.collections.Collections.Collections.collectionIfEmpty\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : Collection<*>, C : R =\n if (isEmpty()) defaultValue() else this\n\n\n/**\n * Checks if all elements in the specified collection are contained in this collection.\n *\n * Allows to overcome type-safety restriction of `containsAll` that requires to pass a collection of type `Collection<E>`.\n * @sample samples.collections.Collections.Collections.collectionContainsAll\n */\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\") // false warning, extension takes precedence in some cases\n@kotlin.internal.InlineOnly\npublic inline fun <@kotlin.internal.OnlyInputTypes T> Collection<T>.containsAll(elements: Collection<T>): Boolean = this.containsAll(elements)\n\n\n/**\n * Returns a new list with the elements of this list randomly shuffled\n * using the specified [random] instance as the source of randomness.\n */\n@SinceKotlin(\"1.3\")\npublic fun <T> Iterable<T>.shuffled(random: Random): List<T> = toMutableList().apply { shuffle(random) }\n\n\ninternal fun <T> List<T>.optimizeReadOnlyList() = when (size) {\n 0 -> emptyList()\n 1 -> listOf(this[0])\n else -> this\n}\n\n/**\n * Searches this list or its range for the provided [element] using the binary search algorithm.\n * The list is expected to be sorted into ascending order according to the Comparable natural ordering of its elements,\n * otherwise the result is undefined.\n *\n * If the list contains multiple elements equal to the specified [element], there is no guarantee which one will be found.\n *\n * `null` value is considered to be less than any non-null value.\n *\n * @return the index of the element, if it is contained in the list within the specified range;\n * otherwise, the inverted insertion point `(-insertion point - 1)`.\n * The insertion point is defined as the index at which the element should be inserted,\n * so that the list (or the specified subrange of list) still remains sorted.\n * @sample samples.collections.Collections.Lists.binarySearchOnComparable\n * @sample samples.collections.Collections.Lists.binarySearchWithBoundaries\n */\npublic fun <T : Comparable<T>> List<T?>.binarySearch(element: T?, fromIndex: Int = 0, toIndex: Int = size): Int {\n rangeCheck(size, fromIndex, toIndex)\n\n var low = fromIndex\n var high = toIndex - 1\n\n while (low <= high) {\n val mid = (low + high).ushr(1) // safe from overflows\n val midVal = get(mid)\n val cmp = compareValues(midVal, element)\n\n if (cmp < 0)\n low = mid + 1\n else if (cmp > 0)\n high = mid - 1\n else\n return mid // key found\n }\n return -(low + 1) // key not found\n}\n\n/**\n * Searches this list or its range for the provided [element] using the binary search algorithm.\n * The list is expected to be sorted into ascending order according to the specified [comparator],\n * otherwise the result is undefined.\n *\n * If the list contains multiple elements equal to the specified [element], there is no guarantee which one will be found.\n *\n * `null` value is considered to be less than any non-null value.\n *\n * @return the index of the element, if it is contained in the list within the specified range;\n * otherwise, the inverted insertion point `(-insertion point - 1)`.\n * The

insertion point is defined as the index at which the element should be inserted,\n * so that the list (or the specified subrange of list) still remains sorted according to the specified [comparator].\n * @sample samples.collections.Collections.Lists.binarySearchWithComparator\n */\npublic fun <T> List<T>.binarySearch(element: T, comparator: Comparator<in T>, fromIndex: Int = 0, toIndex: Int = size): Int {\n rangeCheck(size, fromIndex, toIndex)\n\n var low = fromIndex\n var high = toIndex - 1\n\n while (low <= high) {\n val mid = (low + high).ushr(1) // safe from overflows\n val midVal = get(mid)\n val cmp = comparator.compare(midVal, element)\n\n if (cmp < 0)\n low = mid + 1\n else if (cmp > 0)\n high = mid - 1\n else\n return mid // key found\n }\n return -(low + 1) // key not found\n}\n\n/**\n * Searches this list or its range for an element having the key returned by the specified [selector] function\n * equal to the provided [key] value using the binary search algorithm.\n * The list is expected to be sorted into ascending order according to the Comparable natural ordering of keys of its elements.\n * otherwise the result is undefined.\n *\n * If the list contains multiple elements with the specified [key], there is no guarantee which one will be found.\n *\n * `null` value is considered to be less than any non-null value.\n *\n * @return the index of the element with the specified [key], if it is contained in the list within the specified range;\n * otherwise, the inverted insertion point `(-insertion point - 1)`.\n * The insertion point is defined as the index at which the element should be inserted,\n * so that the list (or the specified subrange of list) still remains sorted.\n * @sample samples.collections.Collections.Lists.binarySearchByKey\n */\npublic inline fun <T, K : Comparable<K>> List<T>.binarySearchBy(\n key: K?,\n fromIndex: Int = 0,\n toIndex: Int = size,\n crossinline selector: (T) -> K?\n): Int =\n binarySearch(fromIndex, toIndex) { compareValues(selector(it), key) }\n\n// do not introduce this overload --- too rare\n//public fun <T, K> List<T>.binarySearchBy(key: K, comparator: Comparator<K>, fromIndex: Int = 0, toIndex: Int = size(), selector: (T) -> K): Int =\n// binarySearch(fromIndex, toIndex) { comparator.compare(selector(it), key) }\n\n\n/**\n * Searches this list or its range for an element for which the given [comparison] function returns zero using the binary search algorithm.\n *\n * The list is expected to be sorted so that the signs of the [comparison] function's return values ascend on the list elements,\n * i.e. negative values come before zero and zeroes come before positive values.\n * Otherwise, the result is undefined.\n *\n * If the list contains multiple elements for which [comparison] returns zero, there is no guarantee which one will be found.\n *\n * @param comparison function that returns zero when called on the list element being searched.\n * On the elements coming before the target element, the function must return negative values;\n * on the elements coming after the target element, the function must return positive values.\n *\n * @return the index of the found element, if it is contained in the list within the specified range;\n * otherwise, the inverted insertion point `(-insertion point - 1)`.\n * The insertion point is defined as the index at which the element should be inserted,\n * so that the list (or the specified subrange of list) still remains sorted.\n * @sample samples.collections.Collections.Lists.binarySearchWithComparisonFunction\n */\npublic fun <T> List<T>.binarySearch(fromIndex: Int = 0, toIndex: Int = size, comparison: (T) -> Int): Int {\n rangeCheck(size, fromIndex, toIndex)\n\n var low = fromIndex\n var high = toIndex - 1\n\n while (low <= high) {\n val mid = (low + high).ushr(1) // safe from overflows\n val midVal = get(mid)\n val cmp = comparison(midVal)\n\n if (cmp < 0)\n low = mid + 1\n else if (cmp > 0)\n high = mid - 1\n else\n return mid // key found\n }\n return -(low + 1) // key not found\n}\n\n/**\n * Checks that `from` and `to` are in\n * the range of [0..size] and throws an appropriate exception, if they aren't.\n */\nprivate fun rangeCheck(size: Int, fromIndex: Int, toIndex: Int) {\n when {\n fromIndex > toIndex -> throw IllegalArgumentException(\"fromIndex ($fromIndex) is greater than toIndex ($toIndex).\")\n fromIndex < 0 -> throw IndexOutOfBoundsException(\"fromIndex ($fromIndex) is less than zero.\")\n toIndex > size -> throw IndexOutOfBoundsException(\"toIndex ($toIndex) is greater than size ($size).\")\n }\n}\n\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\ninternal expect fun checkIndexOverflow(index: Int): Int\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\ninternal expect fun checkCountOverflow(count: Int): Int\n\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\ninternal fun throwIndexOverflow() { throw ArithmeticException(\"Index overflow has happened.\") }\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\ninternal fun throwCountOverflow() { throw ArithmeticException(\"Count overflow has happened.\") }\n\n","/*\n * Copyright

2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"MapsKt\")\n@file:OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n\npackage kotlin.collections\n\nimport kotlin.contracts.*\n\nprivate object EmptyMap : Map<Any?, Nothing>, Serializable {\n   private const val serialVersionUID: Long = 8246714829545688274\n\n   override fun equals(other: Any?): Boolean = other is Map<*, *> && other.isEmpty()\n   override fun hashCode(): Int = 0\n   override fun toString(): String = \"{}\"\n\n   override val size: Int get() = 0\n   override fun isEmpty(): Boolean = true\n\n   override fun containsKey(key: Any?): Boolean = false\n   override fun containsValue(value: Nothing): Boolean = false\n   override fun get(key: Any?): Nothing? = null\n   override val entries: Set<Map.Entry<Any?, Nothing>> get() = EmptySet\n   override val keys: Set<Any?> get() = EmptySet\n   override val values: Collection<Nothing> get() = EmptyList\n\n   private fun readResolve(): Any = EmptyMap\n}\n\n/**\n * Returns an empty read-only map of specified type.\n *\n * The returned map is serializable (JVM).\n * @sample samples.collections.Maps.Instantiation.emptyReadOnlyMap\n */\npublic fun <K, V> emptyMap(): Map<K, V> = @Suppress(\"UNCHECKED_CAST\") (EmptyMap as Map<K, V>)\n\n/**\n * Returns a new read-only map with the specified contents, given as a list of pairs\n * where the first value is the key and the second is the value.\n *\n * If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs.\n *\n * Entries of the map are iterated in the order they were specified.\n *\n * The returned map is serializable (JVM).\n *\n * @sample samples.collections.Maps.Instantiation.mapFromPairs\n */\npublic fun <K, V> mapOf(vararg pairs: Pair<K, V>): Map<K, V> =\n   if (pairs.size > 0) pairs.toMap(LinkedHashMap(mapCapacity(pairs.size))) else emptyMap()\n\n/**\n * Returns an empty read-only map.\n *\n * The returned map is serializable (JVM).\n * @sample samples.collections.Maps.Instantiation.emptyReadOnlyMap\n */\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> mapOf(): Map<K, V> = emptyMap()\n\n/**\n * Returns an empty new [MutableMap].\n *\n * The returned map preserves the entry iteration order.\n * @sample samples.collections.Maps.Instantiation.emptyMutableMap\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> mutableMapOf(): MutableMap<K, V> = LinkedHashMap()\n\n/**\n * Returns a new [MutableMap] with the specified contents, given as a list of pairs\n * where the first component is the key and the second is the value.\n *\n * If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs.\n *\n * Entries of the map are iterated in the order they were specified.\n *\n * @sample samples.collections.Maps.Instantiation.mutableMapFromPairs\n * @sample samples.collections.Maps.Instantiation.emptyMutableMap\n */\npublic fun <K, V> mutableMapOf(vararg pairs: Pair<K, V>): MutableMap<K, V> =\n   LinkedHashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }\n\n/**\n * Returns an empty new [HashMap].\n *\n * @sample samples.collections.Maps.Instantiation.emptyHashMap\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> hashMapOf(): HashMap<K, V> = HashMap<K, V>()\n\n/**\n * Returns a new [HashMap] with the specified contents, given as a list of pairs\n * where the first component is the key and the second is the value.\n *\n * @sample samples.collections.Maps.Instantiation.hashMapFromPairs\n */\npublic fun <K, V> hashMapOf(vararg pairs: Pair<K, V>): HashMap<K, V> = HashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }\n\n/**\n * Returns an empty new [LinkedHashMap].\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> linkedMapOf(): LinkedHashMap<K, V> = LinkedHashMap<K, V>()\n\n/**\n * Returns a new [LinkedHashMap] with the specified contents, given as a list of pairs\n * where the first component is the key and the second is the value.\n *\n * If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs.\n *\n * Entries of the map are iterated in the order they were specified.\n *\n * @sample samples.collections.Maps.Instantiation.linkedMapFromPairs\n */\npublic fun <K, V> linkedMapOf(vararg pairs: Pair<K, V>): LinkedHashMap<K, V> = pairs.toMap(LinkedHashMap(mapCapacity(pairs.size)))\n\n/**\n * Builds a new read-only [Map] by populating a [MutableMap] using the given [builderAction]\n * and returning a read-only map with the same key-value pairs.\n *\n * The map passed as a receiver to the [builderAction] is valid only inside

that function.\n * Using it outside of the function produces an unspecified behavior.\n *\n * Entries of the map are iterated in the order they were added by the [builderAction].\n *\n * The returned map is serializable (JVM).\n *\n * @sample samples.collections.Builders.Maps.buildMapSample\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> buildMap(@BuilderInference builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {\n contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }\n return buildMapInternal(builderAction)\n}\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal expect inline fun <K, V> buildMapInternal(builderAction: MutableMap<K, V>.() -> Unit): Map<K, V>\n\n/**\n * Builds a new read-only [Map] by populating a [MutableMap] using the given [builderAction]\n * and returning a read-only map with the same key-value pairs.\n *\n * The map passed as a receiver to the [builderAction] is valid only inside that function.\n * Using it outside of the function produces an unspecified behavior.\n *\n * [capacity] is used to hint the expected number of pairs added in the [builderAction].\n *\n * Entries of the map are iterated in the order they were added by the [builderAction].\n *\n * The returned map is serializable (JVM).\n *\n * @throws IllegalArgumentException if the given [capacity] is negative.\n *\n * @sample samples.collections.Builders.Maps.buildMapSample\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> buildMap(capacity: Int, @BuilderInference builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {\n contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }\n return buildMapInternal(capacity, builderAction)\n}\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal expect inline fun <K, V> buildMapInternal(capacity: Int, builderAction: MutableMap<K, V>.() -> Unit): Map<K, V>\n\n/**\n * Calculate the initial capacity of a map.\n */\n@PublishedApi\ninternal expect fun mapCapacity(expectedSize: Int): Int\n\n/**\n * Returns `true` if this map is not empty.\n * @sample samples.collections.Maps.Usage.mapIsNotEmpty\n */\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.isNotEmpty(): Boolean = !isEmpty()\n\n/**\n * Returns `true` if this nullable map is either null or empty.\n * @sample samples.collections.Maps.Usage.mapIsNullOrEmpty\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>?.isNullOrEmpty(): Boolean {\n contract {\n returns(false) implies (this@isNullOrEmpty != null)\n }\n\n return this == null || isEmpty()\n}\n\n/**\n * Returns the [Map] if its not `null`, or the empty [Map] otherwise.\n *\n * @sample samples.collections.Maps.Usage.mapOrEmpty\n */\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<K, V>?.orEmpty(): Map<K, V> = this ?: emptyMap()\n\n/**\n * Returns this map if it's not empty\n * or the result of calling [defaultValue] function if the map is empty.\n *\n * @sample samples.collections.Maps.Usage.mapIfEmpty\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun <M, R> M.ifEmpty(defaultValue: () -> R): R where M : Map<*, *>, M : R =\n if (isEmpty()) defaultValue() else this\n\n/**\n * Checks if the map contains the given key.\n *\n * This method allows to use the `x in map` syntax for checking whether an object is contained in the map.\n *\n * @sample samples.collections.Maps.Usage.containsKey\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <@kotlin.internal.OnlyInputTypes K, V> Map<out K, V>.contains(key: K): Boolean = containsKey(key)\n\n/**\n * Returns the value corresponding to the given [key], or `null` if such a key is not present in the map.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <@kotlin.internal.OnlyInputTypes K, V> Map<out K, V>.get(key: K): V? =\n @Suppress(\"UNCHECKED_CAST\") (this as Map<K, V>).get(key)\n\n/**\n * Allows to use the index operator for storing values in a mutable map.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<K, V>.set(key: K, value: V): Unit {\n put(key, value)\n}\n\n/**\n * Returns `true` if the map contains the specified [key].\n *\n * Allows to overcome type-safety restriction of `containsKey` that requires to pass a key of type `K`.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <@kotlin.internal.OnlyInputTypes K> Map<out K, *>.containsKey(key: K): Boolean =\n @Suppress(\"UNCHECKED_CAST\") (this as Map<K, *>).containsKey(key)\n\n/**\n * Returns `true` if the map maps one or more keys to the specified [value].\n *\n * Allows to overcome type-safety restriction of

`containsValue` that requires to pass a value of type `V`.\n *\n * @sample samples.collections.Maps.Usage.containsValue\n */\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\") // false warning, extension takes precedence in some cases\n@kotlin.internal.InlineOnly\npublic inline fun <K, @kotlin.internal.OnlyInputTypes V> Map<K, V>.containsValue(value: V): Boolean = this.containsValue(value)\n\n\n/**\n * Removes the specified key and its corresponding value from this map.\n *\n * @return the previous value associated with the key, or `null` if the key was not present in the map.\n\n * Allows to overcome type-safety restriction of `remove` that requires to pass a key of type `K`.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <@kotlin.internal.OnlyInputTypes K, V> MutableMap<out K, V>.remove(key: K): V? =\n   @Suppress(\"UNCHECKED_CAST\") (this as MutableMap<K, V>).remove(key)\n\n/**\n * Returns the key component of the map entry.\n *\n * This method allows to use destructuring declarations when working with maps, for example:\n * ```\n * for ((key, value) in map) {\n *   // do something with the key and the value\n * }\n * ```\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> Map.Entry<K, V>.component1(): K = key\n\n/**\n * Returns the value component of the map entry.\n *\n * This method allows to use destructuring declarations when working with maps, for example:\n * ```\n * for ((key, value) in map) {\n *   // do something with the key and the value\n * }\n * ```\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> Map.Entry<K, V>.component2(): V = value\n\n/**\n * Converts entry to [Pair] with key being first component and value being second.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map.Entry<K, V>.toPair(): Pair<K, V> = Pair(key, value)\n\n/**\n * Returns the value for the given key, or the result of the [defaultValue] function if there was no entry for the given key.\n *\n * @sample samples.collections.Maps.Usage.getOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<K, V>.getOrElse(key: K, defaultValue: () -> V): V = get(key) ?: defaultValue()\n\n\ninternal inline fun <K, V> Map<K, V>.getOrElseNullable(key: K, defaultValue: () -> V): V {\n   val value = get(key)\n   if (value == null && !containsKey(key)) {\n   return defaultValue()\n   } else {\n   @Suppress(\"UNCHECKED_CAST\")\n   return value as V\n   }\n}\n\n/**\n * Returns the value for the given [key] or throws an exception if there is no such key in the map.\n *\n * If the map was created by [withDefault], resorts to its `defaultValue` provider function\n * instead of throwing an exception.\n *\n * @throws NoSuchElementException when the map doesn't contain a value for the specified key and\n * no implicit default value was provided for that map.\n */\n@SinceKotlin(\"1.1\")\npublic fun <K, V> Map<K, V>.getValue(key: K): V = getOrImplicitDefault(key)\n\n/**\n * Returns the value for the given key. If the key is not found in the map, calls the [defaultValue] function,\n * puts its result into the map under the given key and returns it.\n *\n * Note that the operation is not guaranteed to be atomic if the map is being modified concurrently.\n *\n * @sample samples.collections.Maps.Usage.getOrPut\n */\npublic inline fun <K, V> MutableMap<K, V>.getOrPut(key: K, defaultValue: () -> V): V {\n   val value = get(key)\n   return if (value == null) {\n   val answer = defaultValue()\n   put(key, answer)\n   answer\n   } else {\n   value\n   }\n}\n\n/**\n * Returns an [Iterator] over the entries in the [Map].\n *\n * @sample samples.collections.Maps.Usage.forOverEntries\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> Map<out K, V>.iterator(): Iterator<Map.Entry<K, V>> = entries.iterator()\n\n/**\n * Returns a [MutableIterator] over the mutable entries in the [MutableMap].\n *\n */\n@kotlin.jvm.JvmName(\"mutableIterator\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<K, V>.iterator(): MutableIterator<MutableMap.MutableEntry<K, V>> = entries.iterator()\n\n/**\n * Populates the given [destination] map with entries having the keys of this map and the values obtained\n * by applying the [transform] function to each entry in this [Map].\n */\npublic inline fun <K, V, R, M : MutableMap<in K, in R>> Map<out K, V>.mapValuesTo(destination: M, transform: (Map.Entry<K, V>) -> R): M {\n   return entries.associateByTo(destination, { it.key }, transform)\n}\n\n/**\n * Populates the given [destination] map with entries having the keys obtained\n * by applying the [transform] function to each entry in this [Map] and the values of this map.\n *\n * In case if any two entries are mapped to the equal keys, the value of the latter one will overwrite\n * the value associated with the former one.\n */\npublic inline fun <K, V, R, M : MutableMap<in R, in V>> Map<out K, V>.mapKeysTo(destination: M, transform: (Map.Entry<K, V>) -> R): M {\n   return entries.associateByTo(destination, transform, { it.value })\n}\n\n/**\n * Puts all the given [pairs] into

this [MutableMap] with the first component in the pair being the key and the second the value.\n */\npublic fun <K, V> MutableMap<in K, in V>.putAll(pairs: Array<out Pair<K, V>>): Unit {\n for ((key, value) in pairs) {\n put(key, value)\n }\n}\n\n/**\n * Puts all the elements of the given collection into this [MutableMap] with the first component in the pair being the key and the second the value.\n */\npublic fun <K, V> MutableMap<in K, in V>.putAll(pairs: Iterable<Pair<K, V>>): Unit {\n for ((key, value) in pairs) {\n put(key, value)\n }\n}\n\n/**\n * Puts all the elements of the given sequence into this [MutableMap] with the first component in the pair being the key and the second the value.\n */\npublic fun <K, V> MutableMap<in K, in V>.putAll(pairs: Sequence<Pair<K, V>>): Unit {\n for ((key, value) in pairs) {\n put(key, value)\n }\n}\n\n/**\n * Returns a new map with entries having the keys of this map and the values obtained by applying the [transform]\n * function to each entry in this [Map].\n *\n * The returned map preserves the entry iteration order of the original map.\n *\n * @sample samples.collections.Maps.Transformations.mapValues\n */\npublic inline fun <K, V, R> Map<out K, V>.mapValues(transform: (Map.Entry<K, V>) -> R): Map<K, R> {\n return mapValuesTo(LinkedHashMap<K, R>(mapCapacity(size)), transform) // .optimizeReadOnlyMap()\n}\n\n/**\n * Returns a new Map with entries having the keys obtained by applying the [transform] function to each entry in this\n * [Map] and the values of this map.\n *\n * In case if any two entries are mapped to the equal keys, the value of the latter one will overwrite\n * the value associated with the former one.\n *\n * The returned map preserves the entry iteration order of the original map.\n *\n * @sample samples.collections.Maps.Transformations.mapKeys\n */\npublic inline fun <K, V, R> Map<out K, V>.mapKeys(transform: (Map.Entry<K, V>) -> R): Map<R, V> {\n return mapKeysTo(LinkedHashMap<R, V>(mapCapacity(size)), transform) // .optimizeReadOnlyMap()\n}\n\n/**\n * Returns a map containing all key-value pairs with keys matching the given [predicate].\n *\n * The returned map preserves the entry iteration order of the original map.\n * @sample samples.collections.Maps.Filtering.filterKeys\n */\npublic inline fun <K, V> Map<out K, V>.filterKeys(predicate: (K) -> Boolean): Map<K, V> {\n val result = LinkedHashMap<K, V>()\n for (entry in this) {\n if (predicate(entry.key)) {\n result.put(entry.key, entry.value)\n }\n }\n return result\n}\n\n/**\n * Returns a map containing all key-value pairs with values matching the given [predicate].\n *\n * The returned map preserves the entry iteration order of the original map.\n * @sample samples.collections.Maps.Filtering.filterValues\n */\npublic inline fun <K, V> Map<out K, V>.filterValues(predicate: (V) -> Boolean): Map<K, V> {\n val result = LinkedHashMap<K, V>()\n for (entry in this) {\n if (predicate(entry.value)) {\n result.put(entry.key, entry.value)\n }\n }\n return result\n}\n\n/**\n * Appends all entries matching the given [predicate] into the mutable map given as [destination] parameter.\n *\n * @return the destination map.\n * @sample samples.collections.Maps.Filtering.filterTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> Map<out K, V>.filterTo(destination: M, predicate: (Map.Entry<K, V>) -> Boolean): M {\n for (element in this) {\n if (predicate(element)) {\n destination.put(element.key, element.value)\n }\n }\n return destination\n}\n\n/**\n * Returns a new map containing all key-value pairs matching the given [predicate].\n *\n * The returned map preserves the entry iteration order of the original map.\n * @sample samples.collections.Maps.Filtering.filter\n */\npublic inline fun <K, V> Map<out K, V>.filter(predicate: (Map.Entry<K, V>) -> Boolean): Map<K, V> {\n return filterTo(LinkedHashMap<K, V>(), predicate)\n}\n\n/**\n * Appends all entries not matching the given [predicate] into the given [destination].\n *\n * @return the destination map.\n * @sample samples.collections.Maps.Filtering.filterNotTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> Map<out K, V>.filterNotTo(destination: M, predicate: (Map.Entry<K, V>) -> Boolean): M {\n for (element in this) {\n if (!predicate(element)) {\n destination.put(element.key, element.value)\n }\n }\n return destination\n}\n\n/**\n * Returns a new map containing all key-value pairs not matching the given [predicate].\n *\n * The returned map preserves the entry iteration order of the original map.\n * @sample samples.collections.Maps.Filtering.filterNot\n */\npublic inline fun <K, V> Map<out K, V>.filterNot(predicate: (Map.Entry<K, V>) -> Boolean): Map<K, V> {\n return filterNotTo(LinkedHashMap<K, V>(), predicate)\n}\n\n/**\n * Returns a new map containing all key-value pairs from the given collection of pairs.\n *\n * The returned map preserves the entry iteration order of the original collection.\n * If any of two pairs would have the same key the last one gets added to the map.\n */\npublic fun <K, V> Iterable<Pair<K, V>>.toMap(): Map<K, V>

{\n   if (this is Collection) {\n   return when (size) {\n   0 -> emptyMap()\n   1 -> mapOf(if (this is List) this[0] else iterator().next())\n   else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))\n   }\n   }\n   return toMap(LinkedHashMap<K, V>()).optimizeReadOnlyMap()\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs from the given collection of pairs.\n */\npublic fun <K, V, M : MutableMap<in K, in V>> Iterable<Pair<K, V>>.toMap(destination: M): M =\n   destination.apply { putAll(this@toMap) }\n\n/**\n * Returns a new map containing all key-value pairs from the given array of pairs.\n *\n * The returned map preserves the entry iteration order of the original array.\n * If any of two pairs would have the same key the last one gets added to the map.\n */\npublic fun <K, V> Array<out Pair<K, V>>.toMap(): Map<K, V> = when (size) {\n   0 -> emptyMap()\n   1 -> mapOf(this[0])\n   else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs from the given array of pairs.\n */\npublic fun <K, V, M : MutableMap<in K, in V>> Array<out Pair<K, V>>.toMap(destination: M): M =\n   destination.apply { putAll(this@toMap) }\n\n/**\n * Returns a new map containing all key-value pairs from the given sequence of pairs.\n *\n * The returned map preserves the entry iteration order of the original sequence.\n * If any of two pairs would have the same key the last one gets added to the map.\n */\npublic fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>()).optimizeReadOnlyMap()\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs from the given sequence of pairs.\n */\npublic fun <K, V, M : MutableMap<in K, in V>> Sequence<Pair<K, V>>.toMap(destination: M): M =\n   destination.apply { putAll(this@toMap) }\n\n/**\n * Returns a new read-only map containing all key-value pairs from the original map.\n *\n * The returned map preserves the entry iteration order of the original map.\n */\n@SinceKotlin(\"1.1\")\npublic fun <K, V> Map<out K, V>.toMap(): Map<K, V> = when (size) {\n   0 -> emptyMap()\n   1 -> toSingletonMap()\n   else -> toMutableMap()\n}\n\n/**\n * Returns a new mutable map containing all key-value pairs from the original map.\n *\n * The returned map preserves the entry iteration order of the original map.\n */\n@SinceKotlin(\"1.1\")\npublic fun <K, V> Map<out K, V>.toMutableMap(): MutableMap<K, V> = LinkedHashMap(this)\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs from the given map.\n */\n@SinceKotlin(\"1.1\")\npublic fun <K, V, M : MutableMap<in K, in V>> Map<out K, V>.toMap(destination: M): M =\n   destination.apply { putAll(this@toMap) }\n\n/**\n * Creates a new read-only map by replacing or adding an entry to this map from a given key-value [pair].\n *\n * The returned map preserves the entry iteration order of the original map.\n * The [pair] is iterated in the end if it has a unique key.\n */\npublic operator fun <K, V> Map<out K, V>.plus(pair: Pair<K, V>): Map<K, V> =\n   if (this.isEmpty()) mapOf(pair) else LinkedHashMap(this).apply { put(pair.first, pair.second) }\n\n/**\n * Creates a new read-only map by replacing or adding entries to this map from a given collection of key-value [pairs].\n *\n * The returned map preserves the entry iteration order of the original map.\n * Those [pairs] with unique keys are iterated in the end in the order of [pairs] collection.\n */\npublic operator fun <K, V> Map<out K, V>.plus(pairs: Iterable<Pair<K, V>>): Map<K, V> =\n   if (this.isEmpty()) pairs.toMap() else LinkedHashMap(this).apply { putAll(pairs) }\n\n/**\n * Creates a new read-only map by replacing or adding entries to this map from a given array of key-value [pairs].\n *\n * The returned map preserves the entry iteration order of the original map.\n * Those [pairs] with unique keys are iterated in the end in the order of [pairs] array.\n */\npublic operator fun <K, V> Map<out K, V>.plus(pairs: Array<out Pair<K, V>>): Map<K, V> =\n   if (this.isEmpty()) pairs.toMap() else LinkedHashMap(this).apply { putAll(pairs) }\n\n/**\n * Creates a new read-only map by replacing or adding entries to this map from a given sequence of key-value [pairs].\n *\n * The returned map preserves the entry iteration order of the original map.\n * Those [pairs] with unique keys are iterated in the end in the order of [pairs] sequence.\n */\npublic operator fun <K, V> Map<out K, V>.plus(pairs: Sequence<Pair<K, V>>): Map<K, V> =\n   LinkedHashMap(this).apply { putAll(pairs) }.optimizeReadOnlyMap()\n\n/**\n * Creates a new read-only map by replacing or adding entries to this map from another [map].\n *\n * The returned map preserves the entry iteration order of the original map.\n * Those entries of another [map] that are missing in this map are iterated in the end in the order of that [map].\n */\npublic operator fun <K, V> Map<out K, V>.plus(map: Map<out K, V>): Map<K, V> =\n   LinkedHashMap(this).apply { putAll(map) }\n\n\n/**\n * Appends or replaces the given [pair] in this mutable map.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V>

MutableMap<in K, in V>.plusAssign(pair: Pair<K, V>) {\n   put(pair.first, pair.second)\n}\n\n/**\n * Appends or replaces all pairs from the given collection of [pairs] in this mutable map.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Iterable<Pair<K, V>>) {\n putAll(pairs)\n}\n\n/**\n * Appends or replaces all pairs from the given array of [pairs] in this mutable map.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Array<out Pair<K, V>>) {\n   putAll(pairs)\n}\n\n/**\n * Appends or replaces all pairs from the given sequence of [pairs] in this mutable map.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Sequence<Pair<K, V>>) {\n   putAll(pairs)\n}\n\n/**\n * Appends or replaces all entries from the given [map] in this mutable map.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(map: Map<K, V>) {\n   putAll(map)\n}\n\n/**\n * Returns a map containing all entries of the original map except the entry with the given [key].\n *\n * The returned map preserves the entry iteration order of the original map.\n */\n@SinceKotlin(\"1.1\")\npublic operator fun <K, V> Map<out K, V>.minus(key: K): Map<K, V> =\n   this.toMutableMap().apply { minusAssign(key) }.optimizeReadOnlyMap()\n\n/**\n * Returns a map containing all entries of the original map except those entries\n * the keys of which are contained in the given [keys] collection.\n *\n * The returned map preserves the entry iteration order of the original map.\n */\n@SinceKotlin(\"1.1\")\npublic operator fun <K, V> Map<out K, V>.minus(keys: Iterable<K>): Map<K, V> =\n   this.toMutableMap().apply { minusAssign(keys) }.optimizeReadOnlyMap()\n\n/**\n * Returns a map containing all entries of the original map except those entries\n * the keys of which are contained in the given [keys] array.\n *\n * The returned map preserves the entry iteration order of the original map.\n */\n@SinceKotlin(\"1.1\")\npublic operator fun <K, V> Map<out K, V>.minus(keys: Array<out K>): Map<K, V> =\n   this.toMutableMap().apply { minusAssign(keys) }.optimizeReadOnlyMap()\n\n/**\n * Returns a map containing all entries of the original map except those entries\n * the keys of which are contained in the given [keys] sequence.\n *\n * The returned map preserves the entry iteration order of the original map.\n */\n@SinceKotlin(\"1.1\")\npublic operator fun <K, V> Map<out K, V>.minus(keys: Sequence<K>): Map<K, V> =\n   this.toMutableMap().apply { minusAssign(keys) }.optimizeReadOnlyMap()\n\n/**\n * Removes the entry with the given [key] from this mutable map.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<K, V>.minusAssign(key: K) {\n   remove(key)\n}\n\n/**\n * Removes all entries the keys of which are contained in the given [keys] collection from this mutable map.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<K, V>.minusAssign(keys: Iterable<K>) {\n   this.keys.removeAll(keys)\n}\n\n/**\n * Removes all entries the keys of which are contained in the given [keys] array from this mutable map.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<K, V>.minusAssign(keys: Array<out K>) {\n   this.keys.removeAll(keys)\n}\n\n/**\n * Removes all entries from the keys of which are contained in the given [keys] sequence from this mutable map.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<K, V>.minusAssign(keys: Sequence<K>) {\n   this.keys.removeAll(keys)\n}\n\n\n// do not expose for now @PublishedApi\ninternal fun <K, V> Map<K, V>.optimizeReadOnlyMap() = when (size) {\n   0 -> emptyMap()\n   1 -> toSingletonMapOrSelf()\n   else -> this\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"SetsKt\")\n@file:OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n\npackage kotlin.collections\n\nimport kotlin.contracts.*\n\ninternal object EmptySet : Set<Nothing>, Serializable {\n   private const val serialVersionUID: Long = 3406603774387020532\n\n   override fun equals(other: Any?): Boolean = other is Set<*> && other.isEmpty()\n   override fun hashCode(): Int = 0\n   override fun toString(): String = \"[]\"\n\n   override val size: Int get() = 0\n   override fun isEmpty(): Boolean = true\n   override fun contains(element: Nothing): Boolean = false\n   override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()\n\n   override fun iterator():

Iterator<Nothing> = EmptyIterator\n\n   private fun readResolve(): Any = EmptySet\n}\n\n\n/**\n * Returns an empty read-only set. The returned set is serializable (JVM).\n * @sample samples.collections.Collections.Sets.emptyReadOnlySet\n */\npublic fun <T> emptySet(): Set<T> = EmptySet\n\n/**\n * Returns a new read-only set with the given elements.\n * Elements of the set are iterated in the order they were specified.\n * The returned set is serializable (JVM).\n * @sample samples.collections.Collections.Sets.readOnlySet\n */\npublic fun <T> setOf(vararg elements: T): Set<T> = if (elements.size > 0) elements.toSet() else emptySet()\n\n/**\n * Returns an empty read-only set. The returned set is serializable (JVM).\n * @sample samples.collections.Collections.Sets.emptyReadOnlySet\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> setOf(): Set<T> = emptySet()\n\n/**\n * Returns an empty new [MutableSet].\n *\n * The returned set preserves the element iteration order.\n * @sample samples.collections.Collections.Sets.emptyMutableSet\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> mutableSetOf(): MutableSet<T> = LinkedHashSet()\n\n/**\n * Returns a new [MutableSet] with the given elements.\n * Elements of the set are iterated in the order they were specified.\n * @sample samples.collections.Collections.Sets.mutableSet\n */\npublic fun <T> mutableSetOf(vararg elements: T): MutableSet<T> = elements.toCollection(LinkedHashSet(mapCapacity(elements.size)))\n\n/** Returns an empty new [HashSet]. */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> hashSetOf(): HashSet<T> = HashSet()\n\n/** Returns a new [HashSet] with the given elements. */\npublic fun <T> hashSetOf(vararg elements: T): HashSet<T> = elements.toCollection(HashSet(mapCapacity(elements.size)))\n\n/**\n * Returns an empty new [LinkedHashSet].\n * @sample samples.collections.Collections.Sets.emptyLinkedHashSet\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> linkedSetOf(): LinkedHashSet<T> = LinkedHashSet()\n\n/**\n * Returns a new [LinkedHashSet] with the given elements.\n * Elements of the set are iterated in the order they were specified.\n * @sample samples.collections.Collections.Sets.linkedHashSet\n */\npublic fun <T> linkedSetOf(vararg elements: T): LinkedHashSet<T> = elements.toCollection(LinkedHashSet(mapCapacity(elements.size)))\n\n/**\n * Returns a new read-only set either with single given element, if it is not null, or empty set if the element is null.\n * The returned set is serializable (JVM).\n * @sample samples.collections.Collections.Sets.setOfNotNull\n */\n@SinceKotlin(\"1.4\")\npublic fun <T : Any> setOfNotNull(element: T?): Set<T> = if (element != null) setOf(element) else emptySet()\n\n/**\n * Returns a new read-only set only with those given elements, that are not null.\n * Elements of the set are iterated in the order they were specified.\n * The returned set is serializable (JVM).\n * @sample samples.collections.Collections.Sets.setOfNotNull\n */\n@SinceKotlin(\"1.4\")\npublic fun <T : Any> setOfNotNull(vararg elements: T?): Set<T> {\n   return elements.filterNotNullTo(LinkedHashSet())\n}\n\n/**\n * Builds a new read-only [Set] by populating a [MutableSet] using the given [builderAction]\n * and returning a read-only set with the same elements.\n *\n * The set passed as a receiver to the [builderAction] is valid only inside that function.\n * Using it outside of the function produces an unspecified behavior.\n *\n * Elements of the set are iterated in the order they were added by the [builderAction].\n *\n * The returned set is serializable (JVM).\n *\n * @sample samples.collections.Builders.Sets.buildSetSample\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <E> buildSet(@BuilderInference builderAction: MutableSet<E>.() -> Unit): Set<E> {\n   contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }\n   return buildSetInternal(builderAction)\n}\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal expect inline fun <E> buildSetInternal(builderAction: MutableSet<E>.() -> Unit): Set<E>\n\n/**\n * Builds a new read-only [Set] by populating a [MutableSet] using the given [builderAction]\n * and returning a read-only set with the same elements.\n *\n * The set passed as a receiver to the [builderAction] is valid only inside that function.\n * Using it outside of the function produces an unspecified behavior.\n *\n * [capacity] is used to hint the expected number of elements added in the [builderAction].\n *\n * Elements of the set are iterated in the order they were added by the [builderAction].\n *\n * The returned set is serializable (JVM).\n *\n * @throws IllegalArgumentException if the given [capacity] is negative.\n *\n * @sample

samples.collections.Builders.Sets.buildSetSample\n
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli
c inline fun <E> buildSet(capacity: Int, @BuilderInference builderAction: MutableSet<E>.() -> Unit): Set<E> {\n
contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }\n   return buildSetInternal(capacity,
builderAction)\n}\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal expect inline
fun <E> buildSetInternal(capacity: Int, builderAction: MutableSet<E>.() -> Unit): Set<E>\n\n\n/** Returns this Set
if it's not `null` and the empty set otherwise. */\n@kotlin.internal.InlineOnly\npublic inline fun <T>
Set<T>?.orEmpty(): Set<T> = this ?: emptySet()\n\ninternal fun <T> Set<T>.optimizeReadOnlySet() = when (size)
{\n   0 -> emptySet()\n   1 -> setOf(iterator().next())\n   else -> this\n}\n","/*\n * Copyright 2010-2018 JetBrains
s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0
license that can be found in the license/LICENSE.txt file.\n
*/\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"StringsKt\")\n@file:Suppress(\"PLATFOR
M_CLASS_MAPPED_TO_KOTLIN\")\n\npackage kotlin.text\n\n/**\n * Parses the string as a signed [Byte]
number and returns the result\n * or `null` if the string is not a valid representation of a number.\n
*/\n@SinceKotlin(\"1.1\")\npublic fun String.toByteOrNull(): Byte? = toByteOrNull(radix = 10)\n\n/**\n * Parses
the string as a signed [Byte] number and returns the result\n * or `null` if the string is not a valid representation of a
number.\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
conversion.\n */\n@SinceKotlin(\"1.1\")\npublic fun String.toByteOrNull(radix: Int): Byte? {\n   val int =
this.toIntOrNull(radix) ?: return null\n   if (int < Byte.MIN_VALUE || int > Byte.MAX_VALUE) return null\n
return int.toByte()\n}\n\n/**\n * Parses the string as a [Short] number and returns the result\n * or `null` if the string
is not a valid representation of a number.\n */\n@SinceKotlin(\"1.1\")\npublic fun String.toShortOrNull(): Short? =
toShortOrNull(radix = 10)\n\n/**\n * Parses the string as a [Short] number and returns the result\n * or `null` if the
string is not a valid representation of a number.\n *\n * @throws IllegalArgumentException when [radix] is not a
valid radix for string to number conversion.\n */\n@SinceKotlin(\"1.1\")\npublic fun String.toShortOrNull(radix:
Int): Short? {\n   val int = this.toIntOrNull(radix) ?: return null\n   if (int < Short.MIN_VALUE || int >
Short.MAX_VALUE) return null\n   return int.toShort()\n}\n\n/**\n * Parses the string as an [Int] number and
returns the result\n * or `null` if the string is not a valid representation of a number.\n
*/\n@SinceKotlin(\"1.1\")\npublic fun String.toIntOrNull(): Int? = toIntOrNull(radix = 10)\n\n/**\n * Parses the
string as an [Int] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n
*\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n
*/\n@SinceKotlin(\"1.1\")\npublic fun String.toIntOrNull(radix: Int): Int? {\n   checkRadix(radix)\n\n   val length
= this.length\n   if (length == 0) return null\n\n   val start: Int\n   val isNegative: Boolean\n   val limit: Int\n\n   val
firstChar = this[0]\n   if (firstChar < '0') { // Possible leading sign\n   if (length == 1) return null // non-digit
(possible sign) only, no digits after\n\n   start = 1\n\n   if (firstChar == '-') {\n   isNegative = true\n
limit = Int.MIN_VALUE\n   } else if (firstChar == '+') {\n   isNegative = false\n   limit = -
Int.MAX_VALUE\n   } else\n   return null\n   } else {\n   start = 0\n   isNegative = false\n   limit
= -Int.MAX_VALUE\n   }\n\n\n   val limitForMaxRadix = (-Int.MAX_VALUE) / 36\n\n   var limitBeforeMul =
limitForMaxRadix\n   var result = 0\n   for (i in start until length) {\n   val digit = digitOf(this[i], radix)\n\n
if (digit < 0) return null\n   if (result < limitBeforeMul) {\n   if (limitBeforeMul == limitForMaxRadix) {\n
limitBeforeMul = limit / radix\n\n   if (result < limitBeforeMul) {\n   return null\n
}\n   } else {\n   return null\n   }\n   }\n\n   result *= radix\n\n   if (result < limit + digit)
return null\n\n   result -= digit\n   }\n\n   return if (isNegative) result else -result\n}\n\n/**\n * Parses the string
as a [Long] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n
*/\n@SinceKotlin(\"1.1\")\npublic fun String.toLongOrNull(): Long? = toLongOrNull(radix = 10)\n\n/**\n * Parses
the string as a [Long] number and returns the result\n * or `null` if the string is not a valid representation of a
number.\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
conversion.\n */\n@SinceKotlin(\"1.1\")\npublic fun String.toLongOrNull(radix: Int): Long? {\n
checkRadix(radix)\n\n   val length = this.length\n   if (length == 0) return null\n\n   val start: Int\n   val isNegative:

```
Boolean\n    val limit: Long\n\n    val firstChar = this[0]\n    if (firstChar < '0') {  // Possible leading sign\n        if (length == 1) return null  // non-digit (possible sign) only, no digits after\n\n        start = 1\n\n        if (firstChar == '-') {\n            isNegative = true\n            limit = Long.MIN_VALUE\n        } else if (firstChar == '+') {\n            isNegative = false\n            limit = -Long.MAX_VALUE\n        } else\n            return null\n    } else {\n        start = 0\n        isNegative = false\n        limit = -Long.MAX_VALUE\n    }\n\n    val limitForMaxRadix = (-Long.MAX_VALUE) / 36\n\n    var limitBeforeMul = limitForMaxRadix\n    var result = 0L\n    for (i in start until length) {\n        val digit = digitOf(this[i], radix)\n\n        if (digit < 0) return null\n        if (result < limitBeforeMul) {\n            if (limitBeforeMul == limitForMaxRadix) {\n                limitBeforeMul = limit / radix\n\n                if (result < limitBeforeMul) {\n                    return null\n                }\n            } else {\n                return null\n            }\n        }\n\n        result *= radix\n\n        if (result < limit + digit) return null\n\n        result -= digit\n    }\n\n    return if (isNegative) result else -result\n}\n\n\ninternal fun numberFormatError(input: String): Nothing = throw NumberFormatException(\"Invalid number format: '$input'\")\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.time\n\nimport kotlin.contracts.*\nimport kotlin.jvm.JvmInline\nimport kotlin.math.*\n\n/**\n * Represents the amount of time one instant of time is away from another instant.\n *\n * A negative duration is possible in a situation when the second instant is earlier than the first one.\n *\n * The type can store duration values up to \u00b1146 years with nanosecond precision,\n * and up to \u00b1146 million years with millisecond precision.\n * If a duration-returning operation provided in `kotlin.time` produces a duration value that doesn't fit into the above range,\n * the returned `Duration` is infinite.\n *\n * An infinite duration value [Duration.INFINITE] can be used to represent infinite timeouts.\n *\n * To construct a duration use either the extension function [toDuration],\n * or the extension properties [hours], [minutes], [seconds], and so on,\n * available on [Int], [Long], and [Double] numeric types.\n *\n * To get the value of this duration expressed in a particular [duration units][DurationUnit]\n * use the functions [toInt], [toLong], and [toDouble]\n * or the properties [inWholeHours], [inWholeMinutes], [inWholeSeconds], [inWholeNanoseconds], and so on.\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalTime::class)\n@JvmInline\npublic value class Duration internal constructor(private val rawValue: Long) : Comparable<Duration> {\n\n    private val value: Long get() = rawValue shr 1\n    private inline val unitDiscriminator: Int get() = rawValue.toInt() and 1\n    private fun isInNanos() = unitDiscriminator == 0\n    private fun isInMillis() = unitDiscriminator == 1\n    private val storageUnit get() = if (isInNanos()) DurationUnit.NANOSECONDS else DurationUnit.MILLISECONDS\n\n    init {\n        if (durationAssertionsEnabled) {\n            if (isInNanos()) {\n                if (value !in -MAX_NANOS..MAX_NANOS) throw AssertionError(\"$value ns is out of nanoseconds range\")\n            } else {\n                if (value !in -MAX_MILLIS..MAX_MILLIS) throw AssertionError(\"$value ms is out of milliseconds range\")\n                if (value in -MAX_NANOS_IN_MILLIS..MAX_NANOS_IN_MILLIS) throw AssertionError(\"$value ms is denormalized\")\n            }\n        }\n    }\n\n    companion object {\n        /** The duration equal to exactly 0 seconds. */\n        public val ZERO: Duration = Duration(0L)\n\n        /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */\n        public val INFINITE: Duration = durationOfMillis(MAX_MILLIS)\n        internal val NEG_INFINITE: Duration = durationOfMillis(-MAX_MILLIS)\n\n        /** Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified [targetUnit]. */\n        @ExperimentalTime\n        public fun convert(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double =\n            convertDurationUnit(value, sourceUnit, targetUnit)\n\n        // Duration construction extension properties in Duration companion scope\n\n        /** Returns a [Duration] equal to this [Int] number of nanoseconds. */\n        @kotlin.internal.InlineOnly\n        public inline val Int.nanoseconds get() = toDuration(DurationUnit.NANOSECONDS)\n\n        /** Returns a [Duration] equal to this [Long] number of nanoseconds. */\n        @kotlin.internal.InlineOnly\n        public inline val Long.nanoseconds get() = toDuration(DurationUnit.NANOSECONDS)\n\n        /**\n         * Returns a [Duration] equal to this [Double] number of nanoseconds.\n         *\n         * Depending on its magnitude, the value is rounded to an integer number of nanoseconds or milliseconds.\n         *\n         * @throws IllegalArgumentException if this [Double] value is `NaN`.\n         */\n        @kotlin.internal.InlineOnly\n        public inline val Double.nanoseconds get() =
```

toDuration(DurationUnit.NANOSECONDS)\n\n\n /** Returns a [Duration] equal to this [Int] number of microseconds. */\n @kotlin.internal.InlineOnly\n public inline val Int.microseconds get() = toDuration(DurationUnit.MICROSECONDS)\n\n /** Returns a [Duration] equal to this [Long] number of microseconds. */\n @kotlin.internal.InlineOnly\n public inline val Long.microseconds get() = toDuration(DurationUnit.MICROSECONDS)\n\n /**\n * Returns a [Duration] equal to this [Double] number of microseconds.\n *\n * Depending on its magnitude, the value is rounded to an integer number of nanoseconds or milliseconds.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n @kotlin.internal.InlineOnly\n public inline val Double.microseconds get() = toDuration(DurationUnit.MICROSECONDS)\n\n\n /** Returns a [Duration] equal to this [Int] number of milliseconds. */\n @kotlin.internal.InlineOnly\n public inline val Int.milliseconds get() = toDuration(DurationUnit.MILLISECONDS)\n\n /** Returns a [Duration] equal to this [Long] number of milliseconds. */\n @kotlin.internal.InlineOnly\n public inline val Long.milliseconds get() = toDuration(DurationUnit.MILLISECONDS)\n\n /**\n * Returns a [Duration] equal to this [Double] number of milliseconds.\n *\n * Depending on its magnitude, the value is rounded to an integer number of nanoseconds or milliseconds.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n @kotlin.internal.InlineOnly\n public inline val Double.milliseconds get() = toDuration(DurationUnit.MILLISECONDS)\n\n\n /** Returns a [Duration] equal to this [Int] number of seconds. */\n @kotlin.internal.InlineOnly\n public inline val Int.seconds get() = toDuration(DurationUnit.SECONDS)\n\n /** Returns a [Duration] equal to this [Long] number of seconds. */\n @kotlin.internal.InlineOnly\n public inline val Long.seconds get() = toDuration(DurationUnit.SECONDS)\n\n /**\n * Returns a [Duration] equal to this [Double] number of seconds.\n *\n * Depending on its magnitude, the value is rounded to an integer number of nanoseconds or milliseconds.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n @kotlin.internal.InlineOnly\n public inline val Double.seconds get() = toDuration(DurationUnit.SECONDS)\n\n\n /** Returns a [Duration] equal to this [Int] number of minutes. */\n @kotlin.internal.InlineOnly\n public inline val Int.minutes get() = toDuration(DurationUnit.MINUTES)\n\n /** Returns a [Duration] equal to this [Long] number of minutes. */\n @kotlin.internal.InlineOnly\n public inline val Long.minutes get() = toDuration(DurationUnit.MINUTES)\n\n /**\n * Returns a [Duration] equal to this [Double] number of minutes.\n *\n * Depending on its magnitude, the value is rounded to an integer number of nanoseconds or milliseconds.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n @kotlin.internal.InlineOnly\n public inline val Double.minutes get() = toDuration(DurationUnit.MINUTES)\n\n\n /** Returns a [Duration] equal to this [Int] number of hours. */\n @kotlin.internal.InlineOnly\n public inline val Int.hours get() = toDuration(DurationUnit.HOURS)\n\n /** Returns a [Duration] equal to this [Long] number of hours. */\n @kotlin.internal.InlineOnly\n public inline val Long.hours get() = toDuration(DurationUnit.HOURS)\n\n /**\n * Returns a [Duration] equal to this [Double] number of hours.\n *\n * Depending on its magnitude, the value is rounded to an integer number of nanoseconds or milliseconds.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n @kotlin.internal.InlineOnly\n public inline val Double.hours get() = toDuration(DurationUnit.HOURS)\n\n\n /** Returns a [Duration] equal to this [Int] number of days. */\n @kotlin.internal.InlineOnly\n public inline val Int.days get() = toDuration(DurationUnit.DAYS)\n\n /** Returns a [Duration] equal to this [Long] number of days. */\n @kotlin.internal.InlineOnly\n public inline val Long.days get() = toDuration(DurationUnit.DAYS)\n\n /**\n * Returns a [Duration] equal to this [Double] number of days.\n *\n * Depending on its magnitude, the value is rounded to an integer number of nanoseconds or milliseconds.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n @kotlin.internal.InlineOnly\n public inline val Double.days get() = toDuration(DurationUnit.DAYS)\n\n\n // deprecated static factory functions\n\n /** Returns a [Duration] representing the specified [value] number of nanoseconds. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Int.nanoseconds' extension property from Duration.Companion

instead.\", ReplaceWith(\"value.nanoseconds\", \"kotlin.time.Duration.Companion.nanoseconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun nanoseconds(value: Int): Duration = value.toDuration(DurationUnit.NANOSECONDS)\n\n /** Returns a [Duration] representing the specified [value] number of nanoseconds. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Long.nanoseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.nanoseconds\", \"kotlin.time.Duration.Companion.nanoseconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun nanoseconds(value: Long): Duration = value.toDuration(DurationUnit.NANOSECONDS)\n\n /**\n * Returns a [Duration] representing the specified [value] number of nanoseconds.\n *\n * @throws IllegalArgumentException if the provided `Double` [value] is `NaN`.\n */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Double.nanoseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.nanoseconds\", \"kotlin.time.Duration.Companion.nanoseconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun nanoseconds(value: Double): Duration = value.toDuration(DurationUnit.NANOSECONDS)\n\n\n /** Returns a [Duration] representing the specified [value] number of microseconds. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Int.microseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.microseconds\", \"kotlin.time.Duration.Companion.microseconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun microseconds(value: Int): Duration = value.toDuration(DurationUnit.MICROSECONDS)\n\n /** Returns a [Duration] representing the specified [value] number of microseconds. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Long.microseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.microseconds\", \"kotlin.time.Duration.Companion.microseconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun microseconds(value: Long): Duration = value.toDuration(DurationUnit.MICROSECONDS)\n\n /**\n * Returns a [Duration] representing the specified [value] number of microseconds.\n *\n * @throws IllegalArgumentException if the provided `Double` [value] is `NaN`.\n */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Double.microseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.microseconds\", \"kotlin.time.Duration.Companion.microseconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun microseconds(value: Double): Duration = value.toDuration(DurationUnit.MICROSECONDS)\n\n\n /** Returns a [Duration] representing the specified [value] number of milliseconds. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Int.milliseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.milliseconds\", \"kotlin.time.Duration.Companion.milliseconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun milliseconds(value: Int): Duration = value.toDuration(DurationUnit.MILLISECONDS)\n\n /** Returns a [Duration] representing the specified [value] number of milliseconds. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Long.milliseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.milliseconds\", \"kotlin.time.Duration.Companion.milliseconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun milliseconds(value: Long): Duration = value.toDuration(DurationUnit.MILLISECONDS)\n\n /**\n * Returns a [Duration] representing the specified [value] number of milliseconds.\n *\n * @throws IllegalArgumentException if the provided `Double` [value] is `NaN`.\n */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Double.milliseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.milliseconds\", \"kotlin.time.Duration.Companion.milliseconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun milliseconds(value: Double): Duration = value.toDuration(DurationUnit.MILLISECONDS)\n\n\n /** Returns a [Duration] representing the specified [value] number of seconds. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Int.seconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.seconds\",

\"kotlin.time.Duration.Companion.seconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun seconds(value: Int): Duration = value.toDuration(DurationUnit.SECONDS)\n\n /** Returns a [Duration] representing the specified [value] number of seconds. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Long.seconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.seconds\", \"kotlin.time.Duration.Companion.seconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun seconds(value: Long): Duration = value.toDuration(DurationUnit.SECONDS)\n\n /**\n * Returns a [Duration] representing the specified [value] number of seconds.\n *\n * @throws IllegalArgumentException if the provided `Double` [value] is `NaN`.\n */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Double.seconds' extension property from Duration.Companion instead.\", ReplaceWith(\"value.seconds\", \"kotlin.time.Duration.Companion.seconds\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun seconds(value: Double): Duration = value.toDuration(DurationUnit.SECONDS)\n\n\n /** Returns a [Duration] representing the specified [value] number of minutes. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Int.minutes' extension property from Duration.Companion instead.\", ReplaceWith(\"value.minutes\", \"kotlin.time.Duration.Companion.minutes\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun minutes(value: Int): Duration = value.toDuration(DurationUnit.MINUTES)\n\n /** Returns a [Duration] representing the specified [value] number of minutes. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Long.minutes' extension property from Duration.Companion instead.\", ReplaceWith(\"value.minutes\", \"kotlin.time.Duration.Companion.minutes\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun minutes(value: Long): Duration = value.toDuration(DurationUnit.MINUTES)\n\n /**\n * Returns a [Duration] representing the specified [value] number of minutes.\n *\n * @throws IllegalArgumentException if the provided `Double` [value] is `NaN`.\n */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Double.minutes' extension property from Duration.Companion instead.\", ReplaceWith(\"value.minutes\", \"kotlin.time.Duration.Companion.minutes\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun minutes(value: Double): Duration = value.toDuration(DurationUnit.MINUTES)\n\n\n /** Returns a [Duration] representing the specified [value] number of hours. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Int.hours' extension property from Duration.Companion instead.\", ReplaceWith(\"value.hours\", \"kotlin.time.Duration.Companion.hours\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun hours(value: Int): Duration = value.toDuration(DurationUnit.HOURS)\n\n /** Returns a [Duration] representing the specified [value] number of hours. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Long.hours' extension property from Duration.Companion instead.\", ReplaceWith(\"value.hours\", \"kotlin.time.Duration.Companion.hours\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun hours(value: Long): Duration = value.toDuration(DurationUnit.HOURS)\n\n /**\n * Returns a [Duration] representing the specified [value] number of hours.\n *\n * @throws IllegalArgumentException if the provided `Double` [value] is `NaN`.\n */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Double.hours' extension property from Duration.Companion instead.\", ReplaceWith(\"value.hours\", \"kotlin.time.Duration.Companion.hours\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun hours(value: Double): Duration = value.toDuration(DurationUnit.HOURS)\n\n\n /** Returns a [Duration] representing the specified [value] number of days. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Int.days' extension property from Duration.Companion instead.\", ReplaceWith(\"value.days\", \"kotlin.time.Duration.Companion.days\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun days(value: Int): Duration = value.toDuration(DurationUnit.DAYS)\n\n /** Returns a [Duration] representing the specified [value] number of days. */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Long.days' extension property from Duration.Companion instead.\", ReplaceWith(\"value.days\", \"kotlin.time.Duration.Companion.days\"))\n

@DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun days(value: Long): Duration = value.toDuration(DurationUnit.DAYS)\n\n /**\n * Returns a [Duration] representing the specified [value] number of days.\n *\n * @throws IllegalArgumentException if the provided `Double` [value] is `NaN`.\n */\n @SinceKotlin(\"1.5\")\n @ExperimentalTime\n @Deprecated(\"Use 'Double.days' extension property from Duration.Companion instead.\", ReplaceWith(\"value.days\", \"kotlin.time.Duration.Companion.days\"))\n @DeprecatedSinceKotlin(warningSince = \"1.6\")\n public fun days(value: Double): Duration = value.toDuration(DurationUnit.DAYS)\n\n /**\n * Parses a string that represents a duration and returns the parsed [Duration] value.\n *\n * The following formats are accepted:\n *\n * - ISO-8601 Duration format, e.g. `P1DT2H3M4.058S`, see [toIsoString] and [parseIsoString].\n * - The format of string returned by the default [Duration.toString] and `toString` in a specific unit,\n * e.g. `10s`, `1h 30m` or `-(1h 30m)`.\n *\n * @throws IllegalArgumentException if the string doesn't represent a duration in any of the supported formats.\n * @sample samples.time.Durations.parse\n */\n public fun parse(value: String): Duration = try {\n parseDuration(value, strictIso = false)\n } catch (e: IllegalArgumentException) {\n throw IllegalArgumentException(\"Invalid duration string format: '$value'.\", e)\n }\n\n /**\n * Parses a string that represents a duration in ISO-8601 format and returns the parsed [Duration] value.\n *\n * @throws IllegalArgumentException if the string doesn't represent a duration in ISO-8601 format.\n * @sample samples.time.Durations.parseIsoString\n */\n public fun parseIsoString(value: String): Duration = try {\n parseDuration(value, strictIso = true)\n } catch (e: IllegalArgumentException) {\n throw IllegalArgumentException(\"Invalid ISO duration string format: '$value'.\", e)\n }\n\n /**\n * Parses a string that represents a duration and returns the parsed [Duration] value,\n * or `null` if the string doesn't represent a duration in any of the supported formats.\n *\n * The following formats are accepted:\n *\n * - ISO-8601 Duration format, e.g. `P1DT2H3M4.058S`, see [toIsoString] and [parseIsoString].\n * - The format of string returned by the default [Duration.toString] and `toString` in a specific unit,\n * e.g. `10s`, `1h 30m` or `-(1h 30m)`.\n * @sample samples.time.Durations.parse\n */\n public fun parseOrNull(value: String): Duration? = try {\n parseDuration(value, strictIso = false)\n } catch (e: IllegalArgumentException) {\n null\n }\n\n /**\n * Parses a string that represents a duration in ISO-8601 format and returns the parsed [Duration] value,\n * or `null` if the string doesn't represent a duration in ISO-8601 format.\n * @sample samples.time.Durations.parseIsoString\n */\n public fun parseIsoStringOrNull(value: String): Duration? = try {\n parseDuration(value, strictIso = true)\n } catch (e: IllegalArgumentException) {\n null\n }\n }\n\n // arithmetic operators\n\n /** Returns the negative of this value. */\n public operator fun unaryMinus(): Duration = durationOf(-value, unitDiscriminator)\n\n /**\n * Returns a duration whose value is the sum of this and [other] duration values.\n *\n * @throws IllegalArgumentException if the operation results in an undefined value for the given arguments,\n * e.g. when adding infinite durations of different sign.\n */\n public operator fun plus(other: Duration): Duration {\n when {\n this.isInfinite() -> {\n if (other.isFinite() || (this.rawValue xor other.rawValue >= 0))\n return this\n else\n throw IllegalArgumentException(\"Summing infinite durations of different signs yields an undefined result.\")\n }\n other.isInfinite() -> return other\n }\n\n return when {\n this.unitDiscriminator == other.unitDiscriminator -> {\n val result = this.value + other.value // never overflows long, but can overflow long63\n when {\n isInNanos() ->\n durationOfNanosNormalized(result)\n else ->\n durationOfMillisNormalized(result)\n }\n }\n this.isInMillis() ->\n addValuesMixedRanges(this.value, other.value)\n else ->\n addValuesMixedRanges(other.value, this.value)\n }\n }\n\n private fun addValuesMixedRanges(thisMillis: Long, otherNanos: Long): Duration {\n val otherMillis = nanosToMillis(otherNanos)\n val resultMillis = thisMillis + otherMillis\n return if (resultMillis in -MAX_NANOS_IN_MILLIS..MAX_NANOS_IN_MILLIS) {\n val otherNanoRemainder = otherNanos - millisToNanos(otherMillis)\n durationOfNanos(millisToNanos(resultMillis) + otherNanoRemainder)\n }

```
else {\n            durationOfMillis(resultMillis.coerceIn(-MAX_MILLIS, MAX_MILLIS))\n        }\n    }\n\n    /**\n
* Returns a duration whose value is the difference between this and [other] duration values.\n     *\n     * @throws
IllegalArgumentException if the operation results in an undefined value for the given arguments,\n     * e.g. when
subtracting infinite durations of the same sign.\n     */\n    public operator fun minus(other: Duration): Duration =
this + (-other)\n\n    /**\n     * Returns a duration whose value is this duration value multiplied by the given [scale]
number.\n     *\n     * @throws IllegalArgumentException if the operation results in an undefined value for the given
arguments,\n     * e.g. when multiplying an infinite duration by zero.\n     */\n    public operator fun times(scale: Int):
Duration {\n        if (isInfinite()) {\n            return when {\n                scale == 0 -> throw
IllegalArgumentException(\"Multiplying infinite duration by zero yields an undefined result.\")\n                scale > 0
-> this\n                else -> -this\n            }\n        }\n        if (scale == 0) return ZERO\n\n        val value = value\n
val result = value * scale\n        return if (isInNanos()) {\n            if (value in (MAX_NANOS /
Int.MIN_VALUE)..(-MAX_NANOS / Int.MIN_VALUE)) {\n                // can't overflow nanos range for any
scale\n                durationOfNanos(result)\n            } else {\n                if (result / scale == value) {\n
durationOfNanosNormalized(result)\n                } else {\n                    val millis = nanosToMillis(value)\n
val remNanos = value - millisToNanos(millis)\n                    val resultMillis = millis * scale\n                    val
totalMillis = resultMillis + nanosToMillis(remNanos * scale)\n                    if (resultMillis / scale == millis &&
totalMillis xor resultMillis >= 0) {\n                        durationOfMillis(totalMillis.coerceIn(-
MAX_MILLIS..MAX_MILLIS))\n                    } else {\n                        if (value.sign * scale.sign > 0) INFINITE
else NEG_INFINITE\n                    }\n                }\n            }\n        } else {\n            if (result / scale == value) {\n
durationOfMillis(result.coerceIn(-MAX_MILLIS..MAX_MILLIS))\n            } else {\n                if (value.sign
* scale.sign > 0) INFINITE else NEG_INFINITE\n            }\n        }\n    }\n\n    /**\n     * Returns a duration whose
value is this duration value multiplied by the given [scale] number.\n     *\n     * The operation may involve rounding
when the result cannot be represented exactly with a [Double] number.\n     *\n     * @throws
IllegalArgumentException if the operation results in an undefined value for the given arguments,\n     * e.g. when
multiplying an infinite duration by zero.\n     */\n    public operator fun times(scale: Double): Duration {\n        val
intScale = scale.roundToInt()\n        if (intScale.toDouble() == scale) {\n            return times(intScale)\n        }\n\n
val unit = storageUnit\n        val result = toDouble(unit) * scale\n        return result.toDuration(unit)\n    }\n\n
/**\n     * Returns a duration whose value is this duration value divided by the given [scale] number.\n     *\n     *
@throws IllegalArgumentException if the operation results in an undefined value for the given arguments,\n     *
e.g. when dividing zero duration by zero.\n     */\n    public operator fun div(scale: Int): Duration {\n        if (scale ==
0) {\n            return when {\n                isPositive() -> INFINITE\n                isNegative() -> NEG_INFINITE\n
else -> throw IllegalArgumentException(\"Dividing zero duration by zero yields an undefined result.\")\n
}\n        }\n        if (isInNanos()) {\n            return durationOfNanos(value / scale)\n        } else {\n            if
(isInfinite())\n                return this * scale.sign\n\n            val result = value / scale\n\n            if (result in -
MAX_NANOS_IN_MILLIS..MAX_NANOS_IN_MILLIS) {\n                val rem = millisToNanos(value - (result *
scale)) / scale\n                return durationOfNanos(millisToNanos(result) + rem)\n            }\n            return
durationOfMillis(result)\n        }\n    }\n\n    /**\n     * Returns a duration whose value is this duration value divided
by the given [scale] number.\n     *\n     * @throws IllegalArgumentException if the operation results in an
undefined value for the given arguments,\n     * e.g. when dividing an infinite duration by infinity or zero duration
by zero.\n     */\n    public operator fun div(scale: Double): Duration {\n        val intScale = scale.roundToInt()\n
if (intScale.toDouble() == scale && intScale != 0) {\n            return div(intScale)\n        }\n\n        val unit =
storageUnit\n        val result = toDouble(unit) / scale\n        return result.toDuration(unit)\n    }\n\n    /** Returns a
number that is the ratio of this and [other] duration values. */\n    public operator fun div(other: Duration): Double
{\n        val coarserUnit = maxOf(this.storageUnit, other.storageUnit)\n        return this.toDouble(coarserUnit) /
other.toDouble(coarserUnit)\n    }\n\n    /** Returns true, if the duration value is less than zero. */\n    public fun
isNegative(): Boolean = rawValue < 0\n\n    /** Returns true, if the duration value is greater than zero. */\n    public
fun isPositive(): Boolean = rawValue > 0\n\n    /** Returns true, if the duration value is infinite. */\n    public fun
isInfinite(): Boolean = rawValue == INFINITE.rawValue || rawValue == NEG_INFINITE.rawValue\n\n    /**
```

Returns true, if the duration value is finite. */\n public fun isFinite(): Boolean = !isInfinite()\n\n /** Returns the absolute value of this value. The returned value is always non-negative. */\n public val absoluteValue: Duration get() = if (isNegative()) -this else this\n\n override fun compareTo(other: Duration): Int {\n val compareBits = this.rawValue xor other.rawValue\n if (compareBits < 0 || compareBits.toInt() and 1 == 0) // different signs or same sign/same range\n return this.rawValue.compareTo(other.rawValue)\n // same sign/different ranges\n val r = this.unitDiscriminator - other.unitDiscriminator // compare ranges\n return if (isNegative()) -r else r\n }\n\n\n // splitting to components\n\n /**\n * Splits this duration into days, hours, minutes, seconds, and nanoseconds and executes the given [action] with these components.\n * The result of [action] is returned as the result of this function.\n *\n * - `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;\n * - `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;\n * - `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;\n * - `hours` represents the whole number of hours in this duration, and its absolute value is less than 24;\n * - `days` represents the whole number of days in this duration.\n *\n * Infinite durations are represented as either [Long.MAX_VALUE] days, or [Long.MIN_VALUE] days (depending on the sign of infinity),\n * and zeroes in the lower components.\n */\n public inline fun <T> toComponents(action: (days: Long, hours: Int, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T {\n contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }\n return action(inWholeDays, hoursComponent, minutesComponent, secondsComponent, nanosecondsComponent)\n }\n\n /**\n * Splits this duration into hours, minutes, seconds, and nanoseconds and executes the given [action] with these components.\n * The result of [action] is returned as the result of this function.\n *\n * - `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;\n * - `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;\n * - `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;\n * - `hours` represents the whole number of hours in this duration.\n *\n * Infinite durations are represented as either [Long.MAX_VALUE] hours, or [Long.MIN_VALUE] hours (depending on the sign of infinity),\n * and zeroes in the lower components.\n */\n public inline fun <T> toComponents(action: (hours: Long, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T {\n contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }\n return action(inWholeHours, minutesComponent, secondsComponent, nanosecondsComponent)\n }\n\n /**\n * Splits this duration into minutes, seconds, and nanoseconds and executes the given [action] with these components.\n * The result of [action] is returned as the result of this function.\n *\n * - `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;\n * - `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;\n * - `minutes` represents the whole number of minutes in this duration.\n *\n * Infinite durations are represented as either [Long.MAX_VALUE] minutes, or [Long.MIN_VALUE] minutes (depending on the sign of infinity),\n * and zeroes in the lower components.\n */\n public inline fun <T> toComponents(action: (minutes: Long, seconds: Int, nanoseconds: Int) -> T): T {\n contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }\n return action(inWholeMinutes, secondsComponent, nanosecondsComponent)\n }\n\n /**\n * Splits this duration into seconds, and nanoseconds and executes the given [action] with these components.\n * The result of [action] is returned as the result of this function.\n *\n * - `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;\n * - `seconds` represents the whole number of seconds in this duration.\n *\n * Infinite durations are represented as either [Long.MAX_VALUE] seconds, or [Long.MIN_VALUE] seconds (depending on the sign of infinity),\n * and zero nanoseconds.\n */\n public inline fun <T> toComponents(action: (seconds: Long, nanoseconds: Int) -> T): T {\n contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }\n return action(inWholeSeconds, nanosecondsComponent)\n }\n\n @PublishedApi\n internal val hoursComponent: Int\n get() = if (isInfinite()) 0 else (inWholeHours % 24).toInt()\n\n @PublishedApi\n internal val minutesComponent: Int\n get() = if (isInfinite()) 0 else (inWholeMinutes % 60).toInt()\n\n @PublishedApi\n internal val secondsComponent: Int\n get() = if (isInfinite()) 0 else (inWholeSeconds % 60).toInt()\n\n

```
@PublishedApi\n    internal val nanosecondsComponent: Int\n        get() = when {\n            isInfinite() -> 0\n            isInMillis() -> millisToNanos(value % 1_000).toInt()\n            else -> (value % 1_000_000_000).toInt()\n        }\n\n\n    // conversion to units\n\n    /**\n     * Returns the value of this duration expressed as a [Double] number of the specified [unit].\n     *\n     * The operation may involve rounding when the result cannot be represented exactly with a [Double] number.\n     *\n     * An infinite duration value is converted either to [Double.POSITIVE_INFINITY] or [Double.NEGATIVE_INFINITY] depending on its sign.\n     */\n    public fun toDouble(unit: DurationUnit): Double {\n        return when (rawValue) {\n            INFINITE.rawValue -> Double.POSITIVE_INFINITY\n            NEG_INFINITE.rawValue -> Double.NEGATIVE_INFINITY\n            else -> {\n                // TODO: whether it's ok to convert to Double before scaling\n                convertDurationUnit(value.toDouble(), storageUnit, unit)\n            }\n        }\n    }\n\n    /**\n     * Returns the value of this duration expressed as a [Long] number of the specified [unit].\n     *\n     * If the result doesn't fit in the range of [Long] type, it is coerced into that range:\n     * - [Long.MIN_VALUE] is returned if it's less than `Long.MIN_VALUE`,\n     * - [Long.MAX_VALUE] is returned if it's greater than `Long.MAX_VALUE`.\n     *\n     * An infinite duration value is converted either to [Long.MAX_VALUE] or [Long.MIN_VALUE] depending on its sign.\n     */\n    public fun toLong(unit: DurationUnit): Long {\n        return when (rawValue) {\n            INFINITE.rawValue -> Long.MAX_VALUE\n            NEG_INFINITE.rawValue -> Long.MIN_VALUE\n            else -> convertDurationUnit(value, storageUnit, unit)\n        }\n    }\n\n    /**\n     * Returns the value of this duration expressed as an [Int] number of the specified [unit].\n     *\n     * If the result doesn't fit in the range of [Int] type, it is coerced into that range:\n     * - [Int.MIN_VALUE] is returned if it's less than `Int.MIN_VALUE`,\n     * - [Int.MAX_VALUE] is returned if it's greater than `Int.MAX_VALUE`.\n     *\n     * An infinite duration value is converted either to [Int.MAX_VALUE] or [Int.MIN_VALUE] depending on its sign.\n     */\n    public fun toInt(unit: DurationUnit): Int =\n        toLong(unit).coerceIn(Int.MIN_VALUE.toLong(), Int.MAX_VALUE.toLong()).toInt()\n\n    /** The value of this duration expressed as a [Double] number of days. */\n    @ExperimentalTime\n    @Deprecated(\"Use inWholeDays property instead or convert toDouble(DAYS) if a double value is required.\", ReplaceWith(\"toDouble(DurationUnit.DAYS)\"))\n    public val inDays: Double get() = toDouble(DurationUnit.DAYS)\n\n    /** The value of this duration expressed as a [Double] number of hours. */\n    @ExperimentalTime\n    @Deprecated(\"Use inWholeHours property instead or convert toDouble(HOURS) if a double value is required.\", ReplaceWith(\"toDouble(DurationUnit.HOURS)\"))\n    public val inHours: Double get() = toDouble(DurationUnit.HOURS)\n\n    /** The value of this duration expressed as a [Double] number of minutes. */\n    @ExperimentalTime\n    @Deprecated(\"Use inWholeMinutes property instead or convert toDouble(MINUTES) if a double value is required.\", ReplaceWith(\"toDouble(DurationUnit.MINUTES)\"))\n    public val inMinutes: Double get() = toDouble(DurationUnit.MINUTES)\n\n    /** The value of this duration expressed as a [Double] number of seconds. */\n    @ExperimentalTime\n    @Deprecated(\"Use inWholeSeconds property instead or convert toDouble(SECONDS) if a double value is required.\", ReplaceWith(\"toDouble(DurationUnit.SECONDS)\"))\n    public val inSeconds: Double get() = toDouble(DurationUnit.SECONDS)\n\n    /** The value of this duration expressed as a [Double] number of milliseconds. */\n    @ExperimentalTime\n    @Deprecated(\"Use inWholeMilliseconds property instead or convert toDouble(MILLISECONDS) if a double value is required.\", ReplaceWith(\"toDouble(DurationUnit.MILLISECONDS)\"))\n    public val inMilliseconds: Double get() = toDouble(DurationUnit.MILLISECONDS)\n\n    /** The value of this duration expressed as a [Double] number of microseconds. */\n    @ExperimentalTime\n    @Deprecated(\"Use inWholeMicroseconds property instead or convert toDouble(MICROSECONDS) if a double value is required.\", ReplaceWith(\"toDouble(DurationUnit.MICROSECONDS)\"))\n    public val inMicroseconds: Double get() = toDouble(DurationUnit.MICROSECONDS)\n\n    /** The value of this duration expressed as a [Double] number of nanoseconds. */\n    @ExperimentalTime\n    @Deprecated(\"Use inWholeNanoseconds property instead or convert toDouble(NANOSECONDS) if a double value is required.\", ReplaceWith(\"toDouble(DurationUnit.NANOSECONDS)\"))\n    public val inNanoseconds: Double get() = toDouble(DurationUnit.NANOSECONDS)\n\n\n    /**\n     * The value of this duration expressed as a [Long]
```

number of days.\n *\n * An infinite duration value is converted either to [Long.MAX_VALUE] or [Long.MIN_VALUE] depending on its sign.\n */\n public val inWholeDays: Long\n get() = toLong(DurationUnit.DAYS)\n\n /**\n * The value of this duration expressed as a [Long] number of hours.\n *\n * An infinite duration value is converted either to [Long.MAX_VALUE] or [Long.MIN_VALUE] depending on its sign.\n */\n public val inWholeHours: Long\n get() = toLong(DurationUnit.HOURS)\n\n /**\n * The value of this duration expressed as a [Long] number of minutes.\n *\n * An infinite duration value is converted either to [Long.MAX_VALUE] or [Long.MIN_VALUE] depending on its sign.\n */\n public val inWholeMinutes: Long\n get() = toLong(DurationUnit.MINUTES)\n\n /**\n * The value of this duration expressed as a [Long] number of seconds.\n *\n * An infinite duration value is converted either to [Long.MAX_VALUE] or [Long.MIN_VALUE] depending on its sign.\n */\n public val inWholeSeconds: Long\n get() = toLong(DurationUnit.SECONDS)\n\n /**\n * The value of this duration expressed as a [Long] number of milliseconds.\n *\n * An infinite duration value is converted either to [Long.MAX_VALUE] or [Long.MIN_VALUE] depending on its sign.\n */\n public val inWholeMilliseconds: Long\n get() {\n return if (isInMillis() && isFinite()) value else toLong(DurationUnit.MILLISECONDS)\n }\n\n /**\n * The value of this duration expressed as a [Long] number of microseconds.\n *\n * If the result doesn't fit in the range of [Long] type, it is coerced into that range:\n * - [Long.MIN_VALUE] is returned if it's less than `Long.MIN_VALUE`,\n * - [Long.MAX_VALUE] is returned if it's greater than `Long.MAX_VALUE`.\n *\n * An infinite duration value is converted either to [Long.MAX_VALUE] or [Long.MIN_VALUE] depending on its sign.\n */\n public val inWholeMicroseconds: Long\n get() = toLong(DurationUnit.MICROSECONDS)\n\n /**\n * The value of this duration expressed as a [Long] number of nanoseconds.\n *\n * If the result doesn't fit in the range of [Long] type, it is coerced into that range:\n * - [Long.MIN_VALUE] is returned if it's less than `Long.MIN_VALUE`,\n * - [Long.MAX_VALUE] is returned if it's greater than `Long.MAX_VALUE`.\n *\n * An infinite duration value is converted either to [Long.MAX_VALUE] or [Long.MIN_VALUE] depending on its sign.\n */\n public val inWholeNanoseconds: Long\n get() {\n val value = value\n return when {\n isInNanos() -> value\n value > Long.MAX_VALUE / NANOS_IN_MILLIS -> Long.MAX_VALUE\n value < Long.MIN_VALUE / NANOS_IN_MILLIS -> Long.MIN_VALUE\n else -> millisToNanos(value)\n }\n }\n\n // shortcuts\n\n /**\n * Returns the value of this duration expressed as a [Long] number of nanoseconds.\n *\n * If the value doesn't fit in the range of [Long] type, it is coerced into that range, see the conversion [Double.toLong] for details.\n *\n * The range of durations that can be expressed as a `Long` number of nanoseconds is approximately \u00b1292 years.\n */\n @ExperimentalTime\n @Deprecated(\"Use inWholeNanoseconds property instead.\", ReplaceWith(\"this.inWholeNanoseconds\"))\n public fun toLongNanoseconds(): Long = inWholeNanoseconds\n\n /**\n * Returns the value of this duration expressed as a [Long] number of milliseconds.\n *\n * The value is coerced to the range of [Long] type, if it doesn't fit in that range, see the conversion [Double.toLong] for details.\n *\n * The range of durations that can be expressed as a `Long` number of milliseconds is approximately \u00b1292 million years.\n */\n @ExperimentalTime\n @Deprecated(\"Use inWholeMilliseconds property instead.\", ReplaceWith(\"this.inWholeMilliseconds\"))\n public fun toLongMilliseconds(): Long = inWholeMilliseconds\n\n /**\n * Returns a string representation of this duration value\n * expressed as a combination of numeric components, each in its own unit.\n *\n * Each component is a number followed by the unit abbreviated name: `d`, `h`, `m`, `s`:\n * `5h`, `1d 12h`, `1h 0m 30.340s`.\n * The last component, usually seconds, can be a number with a fractional part.\n *\n * If the duration is less than a second, it is represented as a single number\n * with one of sub-second units: `ms` (milliseconds), `us` (microseconds), or `ns` (nanoseconds):\n * `140.884ms`, `500us`, `24ns`.\n *\n * A negative duration is prefixed with `-` sign and, if it consists of multiple components, surrounded with parentheses:\n * `-12m` and `-(1h 30m)`.\n *\n * Special cases:\n * - an infinite duration is formatted as `\"Infinity\"` or `\"-Infinity\"` without a unit.\n *\n * It's recommended to use [toIsoString] that uses more strict ISO-8601 format instead of this `toString`\n * when you want to convert a duration to a string in cases of serialization, interchange, etc.\n *\n * @sample samples.time.Durations.toStringDefault\n */\n override fun toString():

```
String = when (rawValue) {\n        0L -> \"0s\"\n        INFINITE.rawValue -> \"Infinity\"\n
NEG_INFINITE.rawValue -> \"-Infinity\"\n        else -> {\n            val isNegative = isNegative()\n
buildString {\n                if (isNegative) append('-')\n                absoluteValue.toComponents { days, hours, minutes,
seconds, nanoseconds ->\n                    val hasDays = days != 0L\n                    val hasHours = hours != 0\n
  val hasMinutes = minutes != 0\n                    val hasSeconds = seconds != 0 || nanoseconds != 0\n                    var
components = 0\n                    if (hasDays) {\n                        append(days).append('d')\n
components++\n                    }\n                    if (hasHours || (hasDays && (hasMinutes || hasSeconds))) {\n
    if (components++ > 0) append(' ')\n                        append(hours).append('h')\n                    }\n                    if
(hasMinutes || (hasSeconds && (hasHours || hasDays))) {\n                        if (components++ > 0) append(' ')\n
        append(minutes).append('m')\n                    }\n                    if (hasSeconds) {\n                        if
(components++ > 0) append(' ')\n                        when {\n                            seconds != 0 || hasDays || hasHours ||
hasMinutes ->\n                                appendFractional(seconds, nanoseconds, 9, \"s\", isZeroes = false)\n
        nanoseconds >= 1_000_000 ->\n                                appendFractional(nanoseconds / 1_000_000, nanoseconds
% 1_000_000, 6, \"ms\", isZeroes = false)\n                            nanoseconds >= 1_000 ->\n
appendFractional(nanoseconds / 1_000, nanoseconds % 1_000, 3, \"us\", isZeroes = false)\n                            else -
>\n                                append(nanoseconds).append(\"ns\")\n                        }\n                    }\n                    if
(isNegative && components > 1) insert(1, '(').append(')')\n                }\n            }\n        }\n    }\n\n    private fun
StringBuilder.appendFractional(whole: Int, fractional: Int, fractionalSize: Int, unit: String, isZeroes: Boolean) {\n
  append(whole)\n        if (fractional != 0) {\n            append('.')\n            val fracString =
fractional.toString().padStart(fractionalSize, '0')\n            val nonZeroDigits = fracString.indexOfLast { it != '0' } +
1\n            when {\n                !isZeroes && nonZeroDigits < 3 -> appendRange(fracString, 0, nonZeroDigits)\n
        else -> appendRange(fracString, 0, ((nonZeroDigits + 2) / 3) * 3)\n            }\n        }\n        append(unit)\n
    }\n\n    /**\n     * Returns a string representation of this duration value expressed in the given [unit]\n     * and
formatted with the specified [decimals] number of digits after decimal point.\n     *\n     * Special cases:\n     *  - an
infinite duration is formatted as `\"Infinity\"` or `\"-Infinity\"` without a unit.\n     *\n     * @param decimals the
number of digits after decimal point to show. The value must be non-negative.\n     * No more than 12 decimals will
be shown, even if a larger number is requested.\n     *\n     * @return the value of duration in the specified [unit]
followed by that unit abbreviated name: `d`, `h`, `m`, `s`, `ms`, `us`, or `ns`.\n     *\n     * @throws
IllegalArgumentException if [decimals] is less than zero.\n     *\n     * @sample
samples.time.Durations.toStringDecimals\n     */\n    public fun toString(unit: DurationUnit, decimals: Int = 0):
String {\n        require(decimals >= 0) { \"decimals must be not negative, but was $decimals\" }\n        val number =
toDouble(unit)\n        if (number.isInfinite()) return number.toString()\n        return formatToExactDecimals(number,
decimals.coerceAtMost(12)) + unit.shortName()\n    }\n\n\n    /**\n     * Returns an ISO-8601 based string
representation of this duration.\n     *\n     * The returned value is presented in the format `PThHmMs.fS`, where `h`,
`m`, `s` are the integer components of this duration (see [toComponents])\n     * and `f` is a fractional part of second.
Depending on the roundness of the value the fractional part can be formatted with either\n     * 0, 3, 6, or 9 decimal
digits.\n     *\n     * The infinite duration is represented as `\"PT9999999999999H\"` which is larger than any
possible finite duration in Kotlin.\n     *\n     * Negative durations are indicated with the sign `-` in the beginning of
the returned string, for example, `\"-PT5M30S\"`.\n     *\n     * @sample samples.time.Durations.toIsoString\n
*/\n    public fun toIsoString(): String = buildString {\n        if (isNegative()) append('-')\n        append(\"PT\")\n
this@Duration.absoluteValue.toComponents { hours, minutes, seconds, nanoseconds ->\n
@Suppress(\"NAME_SHADOWING\")\n            var hours = hours\n            if (isInfinite()) {\n                // use large
enough value instead of Long.MAX_VALUE\n                hours = 9_999_999_999_999\n            }\n            val
hasHours = hours != 0L\n            val hasSeconds = seconds != 0 || nanoseconds != 0\n            val hasMinutes =
minutes != 0 || (hasSeconds && hasHours)\n            if (hasHours) {\n                append(hours).append('H')\n
}\n            if (hasMinutes) {\n                append(minutes).append('M')\n            }\n            if (hasSeconds ||
(!hasHours && !hasMinutes)) {\n                appendFractional(seconds, nanoseconds, 9, \"S\", isZeroes = true)\n
    }\n        }\n    }\n\n}\n\n// constructing from number of units\n// extension functions\n\n/** Returns a [Duration]
```

equal to this [Int] number of the specified [unit].
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalTime::class)\npublic fun Int.toDuration(unit: DurationUnit): Duration {\n   return if (unit <= DurationUnit.SECONDS) {\n durationOfNanos(convertDurationUnitOverflow(this.toLong(), unit, DurationUnit.NANOSECONDS))\n   } else\n   toLong().toDuration(unit)\n}\n\n/** Returns a [Duration] equal to this [Long] number of the specified [unit].
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalTime::class)\npublic fun Long.toDuration(unit: DurationUnit): Duration {\n   val maxNsInUnit = convertDurationUnitOverflow(MAX_NANOS, DurationUnit.NANOSECONDS, unit)\n   if (this in -maxNsInUnit..maxNsInUnit) {\n   return durationOfNanos(convertDurationUnitOverflow(this, unit, DurationUnit.NANOSECONDS))\n   } else {\n   val millis = convertDurationUnit(this, unit, DurationUnit.MILLISECONDS)\n   return durationOfMillis(millis.coerceIn(-MAX_MILLIS, MAX_MILLIS))\n   }\n}\n\n/**\n * Returns a [Duration] equal to this [Double] number of the specified [unit].\n *\n * Depending on its magnitude, the value is rounded to an integer number of nanoseconds or milliseconds.\n *\n * @throws IllegalArgumentException if this `Double` value is `NaN`.\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalTime::class)\npublic fun Double.toDuration(unit: DurationUnit): Duration {\n   val valueInNs = convertDurationUnit(this, unit, DurationUnit.NANOSECONDS)\n   require(!valueInNs.isNaN()) { \"Duration value cannot be NaN.\" }\n   val nanos = valueInNs.roundToLong()\n   return if (nanos in -MAX_NANOS..MAX_NANOS) {\n durationOfNanos(nanos)\n   } else {\n   val millis = convertDurationUnit(this, unit, DurationUnit.MILLISECONDS).roundToLong()\n   durationOfMillisNormalized(millis)\n   }\n}\n\n// constructing from number of units\n// deprecated extension properties\n\n/** Returns a [Duration] equal to this [Int] number of nanoseconds. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Int.nanoseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.nanoseconds\", \"kotlin.time.Duration.Companion.nanoseconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Int.nanoseconds get() = toDuration(DurationUnit.NANOSECONDS)\n\n/** Returns a [Duration] equal to this [Long] number of nanoseconds. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Long.nanoseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.nanoseconds\", \"kotlin.time.Duration.Companion.nanoseconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Long.nanoseconds get() = toDuration(DurationUnit.NANOSECONDS)\n\n/**\n * Returns a [Duration] equal to this [Double] number of nanoseconds.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Double.nanoseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.nanoseconds\", \"kotlin.time.Duration.Companion.nanoseconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Double.nanoseconds get() = toDuration(DurationUnit.NANOSECONDS)\n\n\n/** Returns a [Duration] equal to this [Int] number of microseconds. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Int.microseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.microseconds\", \"kotlin.time.Duration.Companion.microseconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Int.microseconds get() = toDuration(DurationUnit.MICROSECONDS)\n\n/** Returns a [Duration] equal to this [Long] number of microseconds. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Long.microseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.microseconds\", \"kotlin.time.Duration.Companion.microseconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Long.microseconds get() = toDuration(DurationUnit.MICROSECONDS)\n\n/**\n * Returns a [Duration] equal to this [Double] number of microseconds.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Double.microseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.microseconds\", \"kotlin.time.Duration.Companion.microseconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Double.microseconds get() = toDuration(DurationUnit.MICROSECONDS)\n\n\n/** Returns a [Duration] equal to this [Int] number of milliseconds. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Int.milliseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.milliseconds\",

\"kotlin.time.Duration.Companion.milliseconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Int.milliseconds get() = toDuration(DurationUnit.MILLISECONDS)\n\n/** Returns a [Duration] equal to this [Long] number of milliseconds. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Long.milliseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.milliseconds\", \"kotlin.time.Duration.Companion.milliseconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Long.milliseconds get() = toDuration(DurationUnit.MILLISECONDS)\n\n/**\n * Returns a [Duration] equal to this [Double] number of milliseconds.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Double.milliseconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.milliseconds\", \"kotlin.time.Duration.Companion.milliseconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Double.milliseconds get() = toDuration(DurationUnit.MILLISECONDS)\n\n\n/** Returns a [Duration] equal to this [Int] number of seconds. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Int.seconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.seconds\", \"kotlin.time.Duration.Companion.seconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Int.seconds get() = toDuration(DurationUnit.SECONDS)\n\n/** Returns a [Duration] equal to this [Long] number of seconds. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Long.seconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.seconds\", \"kotlin.time.Duration.Companion.seconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Long.seconds get() = toDuration(DurationUnit.SECONDS)\n\n/**\n * Returns a [Duration] equal to this [Double] number of seconds.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Double.seconds' extension property from Duration.Companion instead.\", ReplaceWith(\"this.seconds\", \"kotlin.time.Duration.Companion.seconds\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Double.seconds get() = toDuration(DurationUnit.SECONDS)\n\n\n/** Returns a [Duration] equal to this [Int] number of minutes. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Int.minutes' extension property from Duration.Companion instead.\", ReplaceWith(\"this.minutes\", \"kotlin.time.Duration.Companion.minutes\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Int.minutes get() = toDuration(DurationUnit.MINUTES)\n\n/** Returns a [Duration] equal to this [Long] number of minutes. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Long.minutes' extension property from Duration.Companion instead.\", ReplaceWith(\"this.minutes\", \"kotlin.time.Duration.Companion.minutes\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Long.minutes get() = toDuration(DurationUnit.MINUTES)\n\n/**\n * Returns a [Duration] equal to this [Double] number of minutes.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Double.minutes' extension property from Duration.Companion instead.\", ReplaceWith(\"this.minutes\", \"kotlin.time.Duration.Companion.minutes\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Double.minutes get() = toDuration(DurationUnit.MINUTES)\n\n\n/** Returns a [Duration] equal to this [Int] number of hours. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Int.hours' extension property from Duration.Companion instead.\", ReplaceWith(\"this.hours\", \"kotlin.time.Duration.Companion.hours\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Int.hours get() = toDuration(DurationUnit.HOURS)\n\n/** Returns a [Duration] equal to this [Long] number of hours. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Long.hours' extension property from Duration.Companion instead.\", ReplaceWith(\"this.hours\", \"kotlin.time.Duration.Companion.hours\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Long.hours get() = toDuration(DurationUnit.HOURS)\n\n/**\n * Returns a [Duration] equal to this [Double] number of hours.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Double.hours' extension property from Duration.Companion instead.\", ReplaceWith(\"this.hours\",

```
\"kotlin.time.Duration.Companion.hours\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Double.hours get() = toDuration(DurationUnit.HOURS)\n\n\n/** Returns a [Duration] equal to this [Int] number of days. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Int.days' extension property from Duration.Companion instead.\", ReplaceWith(\"this.days\", \"kotlin.time.Duration.Companion.days\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Int.days get() = toDuration(DurationUnit.DAYS)\n\n/** Returns a [Duration] equal to this [Long] number of days. */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Long.days' extension property from Duration.Companion instead.\", ReplaceWith(\"this.days\", \"kotlin.time.Duration.Companion.days\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Long.days get() = toDuration(DurationUnit.DAYS)\n\n/**\n * Returns a [Duration] equal to this [Double] number of days.\n *\n * @throws IllegalArgumentException if this [Double] value is `NaN`.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\n@Deprecated(\"Use 'Double.days' extension property from Duration.Companion instead.\", ReplaceWith(\"this.days\", \"kotlin.time.Duration.Companion.days\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic val Double.days get() = toDuration(DurationUnit.DAYS)\n\n\n/** Returns a duration whose value is the specified [duration] value multiplied by this number.
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalTime::class)\n@kotlin.internal.InlineOnly\npublic inline operator fun Int.times(duration: Duration): Duration = duration * this\n\n/**\n * Returns a duration whose value is the specified [duration] value multiplied by this number.\n *\n * The operation may involve rounding when the result cannot be represented exactly with a [Double] number.\n *\n * @throws IllegalArgumentException if the operation results in a `NaN` value.\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalTime::class)\n@kotlin.internal.InlineOnly\npublic inline operator fun Double.times(duration: Duration): Duration = duration * this\n\n\n\nprivate fun parseDuration(value: String, strictIso: Boolean): Duration {\n    var length = value.length\n    if (length == 0) throw IllegalArgumentException(\"The string is empty\")\n    var index = 0\n    var result = Duration.ZERO\n    val infinityString = \"Infinity\"\n    when (value[index]) {\n        '+', '-' -> index++\n    }\n    val hasSign = index > 0\n    val isNegative = hasSign && value.startsWith('-')\n    when {\n        length <= index ->\n            throw IllegalArgumentException(\"No components\")\n        value[index] == 'P' -> {\n            if (++index == length) throw IllegalArgumentException()\n            val nonDigitSymbols = \"+-.\"\n            var isTimeComponent = false\n            var prevUnit: DurationUnit? = null\n            while (index < length) {\n                if (value[index] == 'T') {\n                    if (isTimeComponent || ++index == length) throw IllegalArgumentException()\n                    isTimeComponent = true\n                    continue\n                }\n                val component = value.substringWhile(index) { it in '0'..'9' || it in nonDigitSymbols }\n                if (component.isEmpty()) throw IllegalArgumentException()\n                index += component.length\n                val unitChar = value.getOrElse(index) { throw IllegalArgumentException(\"Missing unit for value $component\") }\n                index++\n                val unit = durationUnitByIsoChar(unitChar, isTimeComponent)\n                if (prevUnit != null && prevUnit <= unit) throw IllegalArgumentException(\"Unexpected order of duration components\")\n                prevUnit = unit\n                val dotIndex = component.indexOf('.')\n                if (unit == DurationUnit.SECONDS && dotIndex > 0) {\n                    val whole = component.substring(0, dotIndex)\n                    result += parseOverLongIsoComponent(whole).toDuration(unit)\n                    result += component.substring(dotIndex).toDouble().toDuration(unit)\n                } else {\n                    result += parseOverLongIsoComponent(component).toDuration(unit)\n                }\n            }\n        }\n        strictIso ->\n            throw IllegalArgumentException()\n        value.regionMatches(index, infinityString, 0, length = maxOf(length - index, infinityString.length), ignoreCase = true) -> {\n            result = Duration.INFINITE\n        }\n        else -> {\n            // parse default string format\n            var prevUnit: DurationUnit? = null\n            var afterFirst = false\n            var allowSpaces = !hasSign\n            if (hasSign && value[index] == '(' && value.last() == ')') {\n                allowSpaces = true\n                if (++index == --length) throw IllegalArgumentException(\"No components\")\n            }\n            while (index < length) {\n                if (afterFirst && allowSpaces) {\n                    index =
```

value.skipWhile(index) { it == ' ' }\n }\n afterFirst = true\n val component = value.substringWhile(index) { it in '0'..'9' || it == '.' }\n if (component.isEmpty()) throw IllegalArgumentException()\n index += component.length\n val unitName = value.substringWhile(index) { it in 'a'..'z' }\n index += unitName.length\n val unit = durationUnitByShortName(unitName)\n if (prevUnit != null && prevUnit <= unit) throw IllegalArgumentException(\"Unexpected order of duration components\")\n prevUnit = unit\n val dotIndex = component.indexOf('.')\n if (dotIndex > 0) {\n val whole = component.substring(0, dotIndex)\n result += whole.toLong().toDuration(unit)\n result += component.substring(dotIndex).toDouble().toDuration(unit)\n if (index < length) throw IllegalArgumentException(\"Fractional component must be last\")\n } else {\n result += component.toLong().toDuration(unit)\n }\n }\n }\n return if (isNegative) -result else result\n}\n\n\nprivate fun parseOverLongIsoComponent(value: String): Long {\n val length = value.length\n var startIndex = 0\n if (length > 0 && value[0] in \"+-\") startIndex++\n if ((length - startIndex) > 16 && (startIndex..value.lastIndex).all { value[it] in '0'..'9' }) {\n // all chars are digits, but more than ceiling(log10(MAX_MILLIS / 1000)) of them\n return if (value[0] == '-') Long.MIN_VALUE else Long.MAX_VALUE\n }\n // TODO: replace with just toLong after min JDK becomes 8\n return if (value.startsWith(\"+\")) value.drop(1).toLong() else value.toLong()\n}\n\n\n\nprivate inline fun String.substringWhile(startIndex: Int, predicate: (Char) -> Boolean): String =\n substring(startIndex, skipWhile(startIndex, predicate))\n\nprivate inline fun String.skipWhile(startIndex: Int, predicate: (Char) -> Boolean): Int {\n var i = startIndex\n while (i < length && predicate(this[i])) i++\n return i\n}\n\n\n\n\n\n// The ranges are chosen so that they are:\n// - symmetric relative to zero: this greatly simplifies operations with sign, e.g. unaryMinus and minus.\n// - non-overlapping, but adjacent: the first value that doesn't fit in nanos range, can be exactly represented in millis.\n\ninternal const val NANOS_IN_MILLIS = 1_000_000\n// maximum number duration can store in nanosecond range\ninternal const val MAX_NANOS = Long.MAX_VALUE / 2 / NANOS_IN_MILLIS * NANOS_IN_MILLIS - 1 // ends in ..._999_999\n// maximum number duration can store in millisecond range, also encodes an infinite value\ninternal const val MAX_MILLIS = Long.MAX_VALUE / 2\n// MAX_NANOS expressed in milliseconds\nprivate const val MAX_NANOS_IN_MILLIS = MAX_NANOS / NANOS_IN_MILLIS\n\nprivate fun nanosToMillis(nanos: Long): Long = nanos / NANOS_IN_MILLIS\nprivate fun millisToNanos(millis: Long): Long = millis * NANOS_IN_MILLIS\n\nprivate fun durationOfNanos(normalNanos: Long) = Duration(normalNanos shl 1)\nprivate fun durationOfMillis(normalMillis: Long) = Duration((normalMillis shl 1) + 1)\nprivate fun durationOf(normalValue: Long, unitDiscriminator: Int) = Duration((normalValue shl 1) + unitDiscriminator)\nprivate fun durationOfNanosNormalized(nanos: Long) =\n if (nanos in -MAX_NANOS..MAX_NANOS) {\n durationOfNanos(nanos)\n } else {\n durationOfMillis(nanosToMillis(nanos))\n }\n\nprivate fun durationOfMillisNormalized(millis: Long) =\n if (millis in -MAX_NANOS_IN_MILLIS..MAX_NANOS_IN_MILLIS) {\n durationOfNanos(millisToNanos(millis))\n } else {\n durationOfMillis(millis.coerceIn(-MAX_MILLIS, MAX_MILLIS))\n }\n\ninternal expect val durationAssertionsEnabled: Boolean\n\ninternal expect fun formatToExactDecimals(value: Double, decimals: Int): String\ninternal expect fun formatUpToDecimals(value: Double, decimals: Int): String","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n@file:kotlin.jvm.JvmName(\"UnsignedKt\")\npackage kotlin\n\n@PublishedApi\ninternal fun uintCompare(v1: Int, v2: Int): Int = (v1 xor Int.MIN_VALUE).compareTo(v2 xor Int.MIN_VALUE)\n@PublishedApi\ninternal fun ulongCompare(v1: Long, v2: Long): Int = (v1 xor Long.MIN_VALUE).compareTo(v2 xor Long.MIN_VALUE)\n\n@PublishedApi\ninternal fun uintDivide(v1: UInt, v2: UInt): UInt = (v1.toLong() / v2.toLong()).toUInt()\n@PublishedApi\ninternal fun uintRemainder(v1: UInt, v2: UInt): UInt = (v1.toLong() % v2.toLong()).toUInt()\n\n// Division and remainder are based on Guava's UnsignedLongs implementation\n\n// Copyright 2011 The Guava Authors\n\n@PublishedApi\ninternal fun ulongDivide(v1: ULong, v2: ULong): ULong {\n val dividend =

v1.toLong()\n   val divisor = v2.toLong()\n   if (divisor < 0) { // i.e., divisor >= 2^63:\n   return if (v1 < v2) ULong(0) else ULong(1)\n   }\n\n   // Optimization - use signed division if both dividend and divisor < 2^63\n   if (dividend >= 0) {\n   return ULong(dividend / divisor)\n   }\n\n   // Otherwise, approximate the quotient, check, and correct if necessary.\n   val quotient = ((dividend ushr 1) / divisor) shl 1\n   val rem = dividend - quotient * divisor\n   return ULong(quotient + if (ULong(rem) >= ULong(divisor)) 1 else 0)\n\n}\n\n@PublishedApi\ninternal fun ulongRemainder(v1: ULong, v2: ULong): ULong {\n   val dividend = v1.toLong()\n   val divisor = v2.toLong()\n   if (divisor < 0) { // i.e., divisor >= 2^63:\n   return if (v1 < v2) {\n   v1 // dividend < divisor\n   } else {\n   v1 - v2 // dividend >= divisor\n   }\n   }\n\n   // Optimization - use signed modulus if both dividend and divisor < 2^63\n   if (dividend >= 0) {\n   return ULong(dividend % divisor)\n   }\n\n   // Otherwise, approximate the quotient, check, and correct if necessary.\n   val quotient = ((dividend ushr 1) / divisor) shl 1\n   val rem = dividend - quotient * divisor\n   return ULong(rem - if (ULong(rem) >= ULong(divisor)) divisor else 0)\n}\n\n@PublishedApi\ninternal fun doubleToUInt(v: Double): UInt = when {\n   v.isNaN() -> 0u\n   v <= UInt.MIN_VALUE.toDouble() -> UInt.MIN_VALUE\n   v >= UInt.MAX_VALUE.toDouble() -> UInt.MAX_VALUE\n   v <= Int.MAX_VALUE -> v.toInt().toUInt()\n   else -> (v - Int.MAX_VALUE).toInt().toUInt() + Int.MAX_VALUE.toUInt()   // Int.MAX_VALUE < v < UInt.MAX_VALUE\n}\n\n@PublishedApi\ninternal fun doubleToULong(v: Double): ULong = when {\n   v.isNaN() -> 0u\n   v <= ULong.MIN_VALUE.toDouble() -> ULong.MIN_VALUE\n   v >= ULong.MAX_VALUE.toDouble() -> ULong.MAX_VALUE\n   v < Long.MAX_VALUE -> v.toLong().toULong()\n\n   // Real values from Long.MAX_VALUE to (Long.MAX_VALUE + 1) are not representable in Double, so don't handle them.\n   else -> (v - 9223372036854775808.0).toLong().toULong() + 9223372036854775808uL   // Long.MAX_VALUE + 1 < v < ULong.MAX_VALUE\n}\n\n\n@PublishedApi\ninternal fun uintToDouble(v: Int): Double = (v and Int.MAX_VALUE).toDouble() + (v ushr 31 shl 30).toDouble() * 2\n\n@PublishedApi\ninternal fun ulongToDouble(v: Long): Double = (v ushr 11).toDouble() * 2048 + (v and 2047)\n\n\ninternal fun ulongToString(v: Long): String = ulongToString(v, 10)\n\ninternal fun ulongToString(v: Long, base: Int): String {\n if (v >= 0) return v.toString(base)\n\n   var quotient = ((v ushr 1) / base) shl 1\n   var rem = v - quotient * base\n   if (rem >= base) {\n   rem -= base\n   quotient += 1\n   }\n   return quotient.toString(base) + rem.toString(base)\n}\n\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"CollectionsKt\")\n\npackage kotlin.collections\n\n/**\n * Given an [iterator] function constructs an [Iterable] instance that returns values through the [Iterator]\n * provided by that function.\n * @sample samples.collections.Iterables.Building.iterable\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable(crossinline iterator: () -> Iterator<T>): Iterable<T> = object : Iterable<T> {\n   override fun iterator(): Iterator<T> = iterator()\n}\n\n/**\n * A wrapper over another [Iterable] (or any other object that can produce an [Iterator]) that returns\n * an indexing iterator.\n */\ninternal class IndexingIterable<out T>(private val iteratorFactory: () -> Iterator<T>) : Iterable<IndexedValue<T>> {\n   override fun iterator(): Iterator<IndexedValue<T>> = IndexingIterator(iteratorFactory())\n}\n\n\n/**\n * Returns the size of this iterable if it is known, or `null` otherwise.\n */\n@PublishedApi\ninternal fun <T> Iterable<T>.collectionSizeOrNull(): Int? = if (this is Collection<*>) this.size else null\n\n/**\n * Returns the size of this iterable if it is known, or the specified [default] value otherwise.\n */\n@PublishedApi\ninternal fun <T> Iterable<T>.collectionSizeOrDefault(default: Int): Int = if (this is Collection<*>) this.size else default\n\n\n/**\n * Returns a single list of all elements from all collections in the given collection.\n * @sample samples.collections.Iterables.Operations.flattenIterable\n */\npublic fun <T> Iterable<Iterable<T>>.flatten(): List<T> {\n   val result = ArrayList<T>()\n   for (element in this) {\n   result.addAll(element)\n   }\n   return result\n}\n\n/**\n * Returns a pair of lists, where\n * *first* list is built from the first values of each pair from this collection,\n * *second* list is built from the second values of each pair from this collection.\n * @sample samples.collections.Iterables.Operations.unzipIterable\n */\npublic fun <T, R> Iterable<Pair<T, R>>.unzip():

Pair<List<T>, List<R>> {\n   val expectedSize = collectionSizeOrDefault(10)\n   val listT = ArrayList<T>(expectedSize)\n   val listR = ArrayList<R>(expectedSize)\n   for (pair in this) {\n   listT.add(pair.first)\n   listR.add(pair.second)\n   }\n   return listT to listR\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"SequencesKt\")\n\npackage kotlin.sequences\n\nimport kotlin.random.Random\n\n/**\n * Given an [iterator] function constructs a [Sequence] that returns values through the [Iterator]\n * provided by that function.\n * The values are evaluated lazily, and the sequence is potentially infinite.\n *\n * @sample samples.collections.Sequences.Building.sequenceFromIterator\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence(crossinline iterator: () -> Iterator<T>): Sequence<T> = object : Sequence<T> {\n   override fun iterator(): Iterator<T> = iterator()\n}\n\n/**\n * Creates a sequence that returns all elements from this iterator. The sequence is constrained to be iterated only once.\n *\n * @sample samples.collections.Sequences.Building.sequenceFromIterator\n */\npublic fun <T> Iterator<T>.asSequence(): Sequence<T> = Sequence { this }.constrainOnce()\n\n/**\n * Creates a sequence that returns the specified values.\n *\n * @sample samples.collections.Sequences.Building.sequenceOfValues\n */\npublic fun <T> sequenceOf(vararg elements: T): Sequence<T> = if (elements.isEmpty()) emptySequence() else elements.asSequence()\n\n/**\n * Returns an empty sequence.\n */\npublic fun <T> emptySequence(): Sequence<T> = EmptySequence\n\nprivate object EmptySequence : Sequence<Nothing>, DropTakeSequence<Nothing> {\n   override fun iterator(): Iterator<Nothing> = EmptyIterator\n   override fun drop(n: Int) = EmptySequence\n   override fun take(n: Int) = EmptySequence\n}\n\n/**\n * Returns this sequence if it's not `null` and the empty sequence otherwise.\n * @sample samples.collections.Sequences.Usage.sequenceOrEmpty\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>?.orEmpty(): Sequence<T> = this ?: emptySequence()\n\n\n/**\n * Returns a sequence that iterates through the elements either of this sequence\n * or, if this sequence turns out to be empty, of the sequence returned by [defaultValue] function.\n *\n * @sample samples.collections.Sequences.Usage.sequenceIfEmpty\n */\n@SinceKotlin(\"1.3\")\npublic fun <T> Sequence<T>.ifEmpty(defaultValue: () -> Sequence<T>): Sequence<T> = sequence {\n   val iterator = this@ifEmpty.iterator()\n   if (iterator.hasNext()) {\n   yieldAll(iterator)\n   } else {\n   yieldAll(defaultValue())\n   }\n}\n\n/**\n * Returns a sequence of all elements from all sequences in this sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n *\n * @sample samples.collections.Sequences.Transformations.flattenSequenceOfSequences\n */\npublic fun <T> Sequence<Sequence<T>>.flatten(): Sequence<T> = flatten { it.iterator() }\n\n/**\n * Returns a sequence of all elements from all iterables in this sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n *\n * @sample samples.collections.Sequences.Transformations.flattenSequenceOfLists\n */\n@kotlin.jvm.JvmName(\"flattenSequenceOfIterable\")\npublic fun <T> Sequence<Iterable<T>>.flatten(): Sequence<T> = flatten { it.iterator() }\n\nprivate fun <T, R> Sequence<T>.flatten(iterator: (T) -> Iterator<R>): Sequence<R> {\n   if (this is TransformingSequence<*, *>) {\n   return (this as TransformingSequence<*, T>).flatten(iterator)\n   }\n   return FlatteningSequence(this, { it }, iterator)\n}\n\n/**\n * Returns a pair of lists, where\n * *first* list is built from the first values of each pair from this sequence,\n * *second* list is built from the second values of each pair from this sequence.\n *\n * The operation is _terminal_.\n *\n * @sample samples.collections.Sequences.Transformations.unzip\n */\npublic fun <T, R> Sequence<Pair<T, R>>.unzip(): Pair<List<T>, List<R>> {\n   val listT = ArrayList<T>()\n   val listR = ArrayList<R>()\n   for (pair in this) {\n   listT.add(pair.first)\n   listR.add(pair.second)\n   }\n   return listT to listR\n}\n\n/**\n * Returns a sequence that yields elements of this sequence randomly shuffled.\n *\n * Note that every iteration of the sequence returns elements in a different order.\n *\n * The operation is _intermediate_ and _stateful_.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Sequence<T>.shuffled(): Sequence<T> = shuffled(Random)\n\n/**\n * Returns a sequence that yields elements of this sequence randomly shuffled\n * using the specified [random] instance as the source of randomness.\n *\n * Note that every iteration of the sequence returns elements in a

```
different order.\n *\n * The operation is _intermediate_ and _stateful_.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Sequence<T>.shuffled(random: Random): Sequence<T> = sequence<T> {\n    val buffer = toMutableList()\n    while (buffer.isNotEmpty()) {\n        val j = random.nextInt(buffer.size)\n        val last = buffer.removeLast()\n        val value = if (j < buffer.size) buffer.set(j, last) else last\n        yield(value)\n    }\n}\n\n\n/**\n * A sequence that returns the values from the underlying [sequence] that either match or do not match\n * the specified [predicate].\n *\n * @param sendWhen If `true`, values for which the predicate returns `true` are returned. Otherwise,\n * values for which the predicate returns `false` are returned\n */\ninternal class FilteringSequence<T>(\n    private val sequence: Sequence<T>,\n    private val sendWhen: Boolean = true,\n    private val predicate: (T) -> Boolean\n) : Sequence<T> {\n\n    override fun iterator(): Iterator<T> = object : Iterator<T> {\n        val iterator = sequence.iterator()\n        var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue\n        var nextItem: T? = null\n\n        private fun calcNext() {\n            while (iterator.hasNext()) {\n                val item = iterator.next()\n                if (predicate(item) == sendWhen) {\n                    nextItem = item\n                    nextState = 1\n                    return\n                }\n            }\n            nextState = 0\n        }\n\n        override fun next(): T {\n            if (nextState == -1)\n                calcNext()\n            if (nextState == 0)\n                throw NoSuchElementException()\n            val result = nextItem\n            nextItem = null\n            nextState = -1\n            @Suppress(\"UNCHECKED_CAST\")\n            return result as T\n        }\n\n        override fun hasNext(): Boolean {\n            if (nextState == -1)\n                calcNext()\n            return nextState == 1\n        }\n    }\n}\n\n/**\n * A sequence which returns the results of applying the given [transformer] function to the values\n * in the underlying [sequence].\n */\n\ninternal class TransformingSequence<T, R>\nconstructor(private val sequence: Sequence<T>, private val transformer: (T) -> R) : Sequence<R> {\n    override fun iterator(): Iterator<R> = object : Iterator<R> {\n        val iterator = sequence.iterator()\n        override fun next(): R {\n            return transformer(iterator.next())\n        }\n\n        override fun hasNext(): Boolean {\n            return iterator.hasNext()\n        }\n    }\n\n    internal fun <E> flatten(iterator: (R) -> Iterator<E>): Sequence<E> {\n        return FlatteningSequence<T, R, E>(sequence, transformer, iterator)\n    }\n}\n\n/**\n * A sequence which returns the results of applying the given [transformer] function to the values\n * in the underlying [sequence], where the transformer function takes the index of the value in the underlying\n * sequence along with the value itself.\n */\ninternal class TransformingIndexedSequence<T, R>\nconstructor(private val sequence: Sequence<T>, private val transformer: (Int, T) -> R) : Sequence<R> {\n    override fun iterator(): Iterator<R> = object : Iterator<R> {\n        val iterator = sequence.iterator()\n        var index = 0\n        override fun next(): R {\n            return transformer(checkIndexOverflow(index++), iterator.next())\n        }\n\n        override fun hasNext(): Boolean {\n            return iterator.hasNext()\n        }\n    }\n}\n\n/**\n * A sequence which combines values from the underlying [sequence] with their indices and returns them as\n * [IndexedValue] objects.\n */\ninternal class IndexingSequence<T>\nconstructor(private val sequence: Sequence<T>) : Sequence<IndexedValue<T>> {\n    override fun iterator(): Iterator<IndexedValue<T>> = object : Iterator<IndexedValue<T>> {\n        val iterator = sequence.iterator()\n        var index = 0\n        override fun next(): IndexedValue<T> {\n            return IndexedValue(checkIndexOverflow(index++), iterator.next())\n        }\n\n        override fun hasNext(): Boolean {\n            return iterator.hasNext()\n        }\n    }\n}\n\n/**\n * A sequence which takes the values from two parallel underlying sequences, passes them to the given\n * [transform] function and returns the values returned by that function. The sequence stops returning\n * values as soon as one of the underlying sequences stops returning values.\n */\ninternal class MergingSequence<T1, T2, V>\nconstructor(\n    private val sequence1: Sequence<T1>,\n    private val sequence2: Sequence<T2>,\n    private val transform: (T1, T2) -> V\n) : Sequence<V> {\n    override fun iterator(): Iterator<V> = object : Iterator<V> {\n        val iterator1 = sequence1.iterator()\n        val iterator2 = sequence2.iterator()\n        override fun next(): V {\n            return transform(iterator1.next(), iterator2.next())\n        }\n\n        override fun hasNext(): Boolean {\n            return iterator1.hasNext() && iterator2.hasNext()\n        }\n    }\n}\n\n\ninternal class FlatteningSequence<T, R, E>\nconstructor(\n    private val sequence: Sequence<T>,\n    private val transformer: (T) -> R,\n    private val iterator: (R) -> Iterator<E>\n) : Sequence<E> {\n    override fun iterator(): Iterator<E> = object : Iterator<E> {\n        val iterator = sequence.iterator()\n        var itemIterator: Iterator<E>? = null\n\n        override fun next(): E {\n            if (!ensureItemIterator())\n                throw NoSuchElementException()\n            return itemIterator!!.next()\n
```

```
}\n\n    override fun hasNext(): Boolean {\n        return ensureItemIterator()\n    }\n\n    private fun ensureItemIterator(): Boolean {\n        if (itemIterator?.hasNext() == false)\n            itemIterator = null\n\n        while (itemIterator == null) {\n            if (!iterator.hasNext()) {\n                return false\n            } else {\n                val element = iterator.next()\n                val nextItemIterator = iterator(transformer(element))\n                if (nextItemIterator.hasNext()) {\n                    itemIterator = nextItemIterator\n                    return true\n                }\n            }\n        }\n        return true\n    }\n}\n\ninternal fun <T, C, R> flatMapIndexed(source: Sequence<T>, transform: (Int, T) -> C, iterator: (C) -> Iterator<R>): Sequence<R> =\n    sequence {\n        var index = 0\n        for (element in source) {\n            val result = transform(checkIndexOverflow(index++), element)\n            yieldAll(iterator(result))\n        }\n    }\n\n/**\n * A sequence that supports drop(n) and take(n) operations\n */\ninternal interface DropTakeSequence<T> : Sequence<T> {\n    fun drop(n: Int): Sequence<T>\n    fun take(n: Int): Sequence<T>\n}\n\n/**\n * A sequence that skips [startIndex] values from the underlying [sequence]\n * and stops returning values right before [endIndex], i.e. stops at `endIndex - 1`\n */\ninternal class SubSequence<T>(\n    private val sequence: Sequence<T>,\n    private val startIndex: Int,\n    private val endIndex: Int\n) : Sequence<T>, DropTakeSequence<T> {\n\n    init {\n        require(startIndex >= 0) { \"startIndex should be non-negative, but is $startIndex\" }\n        require(endIndex >= 0) { \"endIndex should be non-negative, but is $endIndex\" }\n        require(endIndex >= startIndex) { \"endIndex should be not less than startIndex, but was $endIndex < $startIndex\" }\n    }\n\n    private val count: Int get() = endIndex - startIndex\n\n    override fun drop(n: Int): Sequence<T> = if (n >= count) emptySequence() else SubSequence(sequence, startIndex + n, endIndex)\n    override fun take(n: Int): Sequence<T> = if (n >= count) this else SubSequence(sequence, startIndex, startIndex + n)\n\n    override fun iterator() = object : Iterator<T> {\n\n        val iterator = sequence.iterator()\n        var position = 0\n\n        // Shouldn't be called from constructor to avoid premature iteration\n        private fun drop() {\n            while (position < startIndex && iterator.hasNext()) {\n                iterator.next()\n                position++\n            }\n        }\n\n        override fun hasNext(): Boolean {\n            drop()\n            return (position < endIndex) && iterator.hasNext()\n        }\n\n        override fun next(): T {\n            drop()\n            if (position >= endIndex)\n                throw NoSuchElementException()\n            position++\n            return iterator.next()\n        }\n    }\n}\n\n/**\n * A sequence that returns at most [count] values from the underlying [sequence], and stops returning values\n * as soon as that count is reached.\n */\ninternal class TakeSequence<T>(\n    private val sequence: Sequence<T>,\n    private val count: Int\n) : Sequence<T>, DropTakeSequence<T> {\n\n    init {\n        require(count >= 0) { \"count must be non-negative, but was $count.\" }\n    }\n\n    override fun drop(n: Int): Sequence<T> = if (n >= count) emptySequence() else SubSequence(sequence, n, count)\n    override fun take(n: Int): Sequence<T> = if (n >= count) this else TakeSequence(sequence, n)\n\n    override fun iterator(): Iterator<T> = object : Iterator<T> {\n        var left = count\n        val iterator = sequence.iterator()\n\n        override fun next(): T {\n            if (left == 0)\n                throw NoSuchElementException()\n            left--\n            return iterator.next()\n        }\n\n        override fun hasNext(): Boolean {\n            return left > 0 && iterator.hasNext()\n        }\n    }\n}\n\n/**\n * A sequence that returns values from the underlying [sequence] while the [predicate] function returns\n * `true`, and stops returning values once the function returns `false` for the next element.\n */\ninternal class TakeWhileSequence<T>\nconstructor(\n    private val sequence: Sequence<T>,\n    private val predicate: (T) -> Boolean\n) : Sequence<T> {\n    override fun iterator(): Iterator<T> = object : Iterator<T> {\n        val iterator = sequence.iterator()\n        var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue\n        var nextItem: T? = null\n\n        private fun calcNext() {\n            if (iterator.hasNext()) {\n                val item = iterator.next()\n                if (predicate(item)) {\n                    nextState = 1\n                    nextItem = item\n                    return\n                }\n            }\n            nextState = 0\n        }\n\n        override fun next(): T {\n            if (nextState == -1)\n                calcNext() // will change nextState\n            if (nextState == 0)\n                throw NoSuchElementException()\n            @Suppress(\"UNCHECKED_CAST\")\n            val result = nextItem as T\n\n            // Clean next to avoid keeping reference on yielded instance\n            nextItem = null\n            nextState = -1\n            return result\n        }\n\n        override fun hasNext(): Boolean {\n            if (nextState == -1)\n                calcNext() // will change nextState\n            return nextState == 1\n        }\n    }\n}\n\n/**\n * A sequence that skips the specified number of values from the underlying [sequence] and returns\n * all values after that.\n */\ninternal class DropSequence<T>(\n    private val
```

```
sequence: Sequence<T>,\n    private val count: Int\n) : Sequence<T>, DropTakeSequence<T> {\n    init {\n require(count >= 0) { \"count must be non-negative, but was $count.\" }\n    }\n\n    override fun drop(n: Int): Sequence<T> = (count + n).let { n1 -> if (n1 < 0) DropSequence(this, n) else DropSequence(sequence, n1) }\n override fun take(n: Int): Sequence<T> = (count + n).let { n1 -> if (n1 < 0) TakeSequence(this, n) else SubSequence(sequence, count, n1) }\n\n    override fun iterator(): Iterator<T> = object : Iterator<T> {\n        val iterator = sequence.iterator()\n        var left = count\n\n        // Shouldn't be called from constructor to avoid premature iteration\n        private fun drop() {\n            while (left > 0 && iterator.hasNext()) {\n iterator.next()\n                left--\n            }\n        }\n\n        override fun next(): T {\n            drop()\n            return iterator.next()\n        }\n\n        override fun hasNext(): Boolean {\n            drop()\n            return iterator.hasNext()\n        }\n    }\n}\n\n/**\n * A sequence that skips the values from the underlying [sequence] while the given [predicate] returns `true` and returns\n * all values after that.\n */\ninternal class DropWhileSequence<T>\nconstructor(\n    private val sequence: Sequence<T>,\n    private val predicate: (T) -> Boolean\n) : Sequence<T> {\n\n    override fun iterator(): Iterator<T> = object : Iterator<T> {\n        val iterator = sequence.iterator()\n        var dropState: Int = -1 // -1 for not dropping, 1 for nextItem, 0 for normal iteration\n var nextItem: T? = null\n\n        private fun drop() {\n            while (iterator.hasNext()) {\n                val item = iterator.next()\n                if (!predicate(item)) {\n                    nextItem = item\n                    dropState = 1\n                    return\n                }\n            }\n            dropState = 0\n        }\n\n        override fun next(): T {\n            if (dropState == -1)\n                drop()\n\n            if (dropState == 1) {\n                @Suppress(\"UNCHECKED_CAST\")\n                val result = nextItem as T\n                nextItem = null\n                dropState = 0\n                return result\n            }\n            return iterator.next()\n        }\n\n        override fun hasNext(): Boolean {\n            if (dropState == -1)\n                drop()\n            return dropState == 1 || iterator.hasNext()\n        }\n    }\n}\n\ninternal class DistinctSequence<T, K>(private val source: Sequence<T>, private val keySelector: (T) -> K) : Sequence<T> {\n    override fun iterator(): Iterator<T> = DistinctIterator(source.iterator(), keySelector)\n}\n\nprivate class DistinctIterator<T, K>(private val source: Iterator<T>, private val keySelector: (T) -> K) : AbstractIterator<T>() {\n    private val observed = HashSet<K>()\n\n    override fun computeNext() {\n        while (source.hasNext()) {\n            val next = source.next()\n            val key = keySelector(next)\n\n            if (observed.add(key)) {\n                setNext(next)\n                return\n            }\n        }\n\n        done()\n    }\n}\n\n\nprivate class GeneratorSequence<T : Any>(private val getInitialValue: () -> T?, private val getNextValue: (T) -> T?) : Sequence<T> {\n    override fun iterator(): Iterator<T> = object : Iterator<T> {\n        var nextItem: T? = null\n        var nextState: Int = -2 // -2 for initial unknown, -1 for next unknown, 0 for done, 1 for continue\n\n        private fun calcNext() {\n            nextItem = if (nextState == -2) getInitialValue() else getNextValue(nextItem!!)\n            nextState = if (nextItem == null) 0 else 1\n        }\n\n        override fun next(): T {\n            if (nextState < 0)\n                calcNext()\n\n            if (nextState == 0)\n                throw NoSuchElementException()\n            val result = nextItem as T\n            // Do not clean nextItem (to avoid keeping reference on yielded instance) -- need to keep state for getNextValue\n            nextState = -1\n            return result\n        }\n\n        override fun hasNext(): Boolean {\n            if (nextState < 0)\n                calcNext()\n            return nextState == 1\n        }\n    }\n}\n\n/**\n * Returns a wrapper sequence that provides values of this sequence, but ensures it can be iterated only one time.\n *\n * The operation is _intermediate_ and _stateless_.\n *\n * [IllegalStateException] is thrown on iterating the returned sequence for the second time and the following times.\n *\n */\npublic fun <T> Sequence<T>.constrainOnce(): Sequence<T> {\n    // as? does not work in js\n    //return this as? ConstrainedOnceSequence<T> ?: ConstrainedOnceSequence(this)\n    return if (this is ConstrainedOnceSequence<T>) this else ConstrainedOnceSequence(this)\n}\n\n\n/**\n * Returns a sequence which invokes the function to calculate the next value on each iteration until the function returns `null`.\n *\n * The returned sequence is constrained to be iterated only once.\n *\n * @see constrainOnce\n * @see kotlin.sequences.sequence\n *\n * @sample samples.collections.Sequences.Building.generateSequence\n */\npublic fun <T : Any> generateSequence(nextFunction: () -> T?): Sequence<T> {\n    return GeneratorSequence(nextFunction, { nextFunction() }).constrainOnce()\n}\n\n/**\n * Returns a sequence defined by the starting value [seed] and the function [nextFunction],\n * which is invoked to calculate the next value based on
```

the previous one on each iteration.\n *\n * The sequence produces values until it encounters first `null` value.\n * If [seed] is `null`, an empty sequence is produced.\n *\n * The sequence can be iterated multiple times, each time starting with [seed].\n *\n * @see kotlin.sequences.sequence\n *\n * @sample samples.collections.Sequences.Building.generateSequenceWithSeed\n */\n@kotlin.internal.LowPriorityInOverloadResolution\npublic fun <T : Any> generateSequence(seed: T?, nextFunction: (T) -> T?): Sequence<T> =\n   if (seed == null)\n   EmptySequence\n   else\n GeneratorSequence({ seed }, nextFunction)\n\n/**\n * Returns a sequence defined by the function [seedFunction], which is invoked to produce the starting value,\n * and the [nextFunction], which is invoked to calculate the next value based on the previous one on each iteration.\n *\n * The sequence produces values until it encounters first `null` value.\n * If [seedFunction] returns `null`, an empty sequence is produced.\n *\n * The sequence can be iterated multiple times.\n *\n * @see kotlin.sequences.sequence\n *\n * @sample samples.collections.Sequences.Building.generateSequenceWithLazySeed\n */\npublic fun <T : Any> generateSequence(seedFunction: () -> T?, nextFunction: (T) -> T?): Sequence<T> =\n GeneratorSequence(seedFunction, nextFunction)\n\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"PreconditionsKt\")\n\npackage kotlin\n\nimport kotlin.contracts.contract\n\n/**\n * Throws an [IllegalArgumentException] if the [value] is false.\n *\n * @sample samples.misc.Preconditions.failRequireWithLazyMessage\n */\n@kotlin.internal.InlineOnly\npublic inline fun require(value: Boolean): Unit {\n   contract {\n   returns() implies value\n   }\n   require(value) { \"Failed requirement.\" }\n}\n\n/**\n * Throws an [IllegalArgumentException] with the result of calling [lazyMessage] if the [value] is false.\n *\n * @sample samples.misc.Preconditions.failRequireWithLazyMessage\n */\n@kotlin.internal.InlineOnly\npublic inline fun require(value: Boolean, lazyMessage: () -> Any): Unit {\n   contract {\n   returns() implies value\n   }\n   if (!value) {\n   val message = lazyMessage()\n   throw IllegalArgumentException(message.toString())\n   }\n}\n\n/**\n * Throws an [IllegalArgumentException] if the [value] is null. Otherwise returns the not null value.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T : Any> requireNotNull(value: T?): T {\n   contract {\n   returns() implies (value != null)\n   }\n   return requireNotNull(value) { \"Required value was null.\" }\n}\n\n/**\n * Throws an [IllegalArgumentException] with the result of calling [lazyMessage] if the [value] is null. Otherwise\n * returns the not null value.\n *\n * @sample samples.misc.Preconditions.failRequireNotNullWithLazyMessage\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T : Any> requireNotNull(value: T?, lazyMessage: () -> Any): T {\n   contract {\n   returns() implies (value != null)\n   }\n\n   if (value == null) {\n   val message = lazyMessage()\n   throw IllegalArgumentException(message.toString())\n   } else {\n   return value\n   }\n}\n\n/**\n * Throws an [IllegalStateException] if the [value] is false.\n *\n * @sample samples.misc.Preconditions.failCheckWithLazyMessage\n */\n@kotlin.internal.InlineOnly\npublic inline fun check(value: Boolean): Unit {\n   contract {\n   returns() implies value\n   }\n   check(value) { \"Check failed.\" }\n}\n\n/**\n * Throws an [IllegalStateException] with the result of calling [lazyMessage] if the [value] is false.\n *\n * @sample samples.misc.Preconditions.failCheckWithLazyMessage\n */\n@kotlin.internal.InlineOnly\npublic inline fun check(value: Boolean, lazyMessage: () -> Any): Unit {\n   contract {\n   returns() implies value\n   }\n   if (!value) {\n   val message = lazyMessage()\n   throw IllegalStateException(message.toString())\n   }\n}\n\n/**\n * Throws an [IllegalStateException] if the [value] is null. Otherwise\n * returns the not null value.\n *\n * @sample samples.misc.Preconditions.failCheckWithLazyMessage\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T : Any> checkNotNull(value: T?): T {\n   contract {\n   returns() implies (value != null)\n   }\n   return checkNotNull(value) { \"Required value was null.\" }\n}\n\n/**\n * Throws an [IllegalStateException] with the result of calling [lazyMessage] if the [value] is null. Otherwise\n * returns the not null value.\n *\n * @sample samples.misc.Preconditions.failCheckWithLazyMessage\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T : Any> checkNotNull(value: T?, lazyMessage: () -> Any): T {\n   contract {\n   returns() implies (value != null)\n   }\n\n   if (value == null) {\n   val message = lazyMessage()\n   throw

IllegalStateException(message.toString())\n   } else {\n   return value\n   }\n}\n\n\n/**\n * Throws an [IllegalStateException] with the given [message].\n *\n * @sample samples.misc.Preconditions.failWithError\n */\n@kotlin.internal.InlineOnly\npublic inline fun error(message: Any): Nothing = throw IllegalStateException(message.toString())\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.js.*\nimport primitiveArrayConcat\nimport withType\nimport kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun <T> Array<out T>.elementAt(index: Int): T {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun ByteArray.elementAt(index: Int): Byte {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun ShortArray.elementAt(index: Int): Short {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun IntArray.elementAt(index: Int): Int {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun LongArray.elementAt(index: Int): Long {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun FloatArray.elementAt(index: Int): Float {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun DoubleArray.elementAt(index: Int): Double {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun BooleanArray.elementAt(index: Int): Boolean {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun CharArray.elementAt(index: Int): Char {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size\") }\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic actual fun <T> Array<out T>.asList(): List<T> {\n   return ArrayList<T>(this.unsafeCast<Array<Any?>>())\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@kotlin.internal.InlineOnly\npublic actual inline fun ByteArray.asList(): List<Byte> {\n   return this.unsafeCast<Array<Byte>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@kotlin.internal.InlineOnly\npublic actual inline fun ShortArray.asList(): List<Short> {\n   return this.unsafeCast<Array<Short>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@kotlin.internal.InlineOnly\npublic actual inline fun IntArray.asList(): List<Int> {\n   return this.unsafeCast<Array<Int>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n

*/\n@kotlin.internal.InlineOnly\npublic actual inline fun LongArray.asList(): List<Long> {\n return this.unsafeCast<Array<Long>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@kotlin.internal.InlineOnly\npublic actual inline fun FloatArray.asList(): List<Float> {\n return this.unsafeCast<Array<Float>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@kotlin.internal.InlineOnly\npublic actual inline fun DoubleArray.asList(): List<Double> {\n return this.unsafeCast<Array<Double>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@kotlin.internal.InlineOnly\npublic actual inline fun BooleanArray.asList(): List<Boolean> {\n return this.unsafeCast<Array<Boolean>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic actual fun CharArray.asList(): List<Char> {\n return object : AbstractList<Char>(), RandomAccess {\n override val size: Int get() = this@asList.size\n override fun isEmpty(): Boolean = this@asList.isEmpty()\n override fun contains(element: Char): Boolean = this@asList.contains(element)\n override fun get(index: Int): Char {\n AbstractList.checkElementIndex(index, size)\n return this@asList[index]\n }\n override fun indexOf(element: Char): Int {\n @Suppress(\"USELESS_CAST\")\n if ((element as Any?) !is Char) return -1\n return this@asList.indexOf(element)\n }\n override fun lastIndexOf(element: Char): Int {\n @Suppress(\"USELESS_CAST\")\n if ((element as Any?) !is Char) return -1\n return this@asList.lastIndexOf(element)\n }\n }\n}\n\n/**\n * Returns `true` if the two specified arrays are *deeply* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * If two corresponding elements are nested arrays, they are also compared deeply.\n * If any of arrays contains itself on any nesting level the behavior is undefined.\n * \n * The elements of other types are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.LowPriorityInOverloadResolution\npublic actual infix fun <T> Array<out T>.contentDeepEquals(other: Array<out T>): Boolean {\n return this.contentDeepEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *deeply* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The specified arrays are also considered deeply equal if both are `null`.\n * \n * If two corresponding elements are nested arrays, they are also compared deeply.\n * If any of arrays contains itself on any nesting level the behavior is undefined.\n * \n * The elements of other types are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayDeepEquals\")\npublic actual infix fun <T> Array<out T>?.contentDeepEquals(other: Array<out T>?): Boolean {\n definedExternally\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n * Nested arrays are treated as lists too.\n * \n * If any of arrays contains itself on any nesting level the behavior is undefined.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.LowPriorityInOverloadResolution\npublic actual fun <T> Array<out T>.contentDeepHashCode(): Int {\n return this.contentDeepHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n * Nested arrays are treated as lists too.\n * \n * If any of arrays contains itself on any nesting level the behavior is undefined.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayDeepHashCode\")\npublic actual fun <T> Array<out T>?.contentDeepHashCode(): Int {\n definedExternally\n}\n\n/**\n * Returns a string representation of the contents of this array as if it is a [List].\n * Nested arrays are treated as lists too.\n * \n * If any of arrays contains itself on any nesting level that reference\n * is rendered as `\"[...]\"` to prevent recursion.\n * \n * @sample samples.collections.Arrays.ContentOperations.contentDeepToString\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.LowPriorityInOverloadResolution\npublic actual fun <T> Array<out T>.contentDeepToString(): String {\n return this.contentDeepToString()\n}\n\n/**\n * Returns a string representation of the contents of this array as if it is a [List].\n * Nested arrays are treated as lists too.\n * \n * If any of arrays contains itself on any nesting level that reference\n * is rendered as `\"[...]\"` to prevent recursion.\n * \n * @sample samples.collections.Arrays.ContentOperations.contentDeepToString\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayDeepToString\")\npublic actual fun <T> Array<out

T>?.contentDeepToString(): String {\n   definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual infix fun <T> Array<out T>.contentEquals(other: Array<out T>): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual infix fun ByteArray.contentEquals(other: ByteArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual infix fun ShortArray.contentEquals(other: ShortArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual infix fun IntArray.contentEquals(other: IntArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual infix fun LongArray.contentEquals(other: LongArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual infix fun FloatArray.contentEquals(other: FloatArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n *

Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual infix fun CharArray.contentEquals(other: CharArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayEquals\")\npublic actual infix fun <T> Array<out T>?.contentEquals(other: Array<out T>?): Boolean {\n   definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayEquals\")\npublic actual infix fun ByteArray?.contentEquals(other: ByteArray?): Boolean {\n   definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayEquals\")\npublic actual infix fun ShortArray?.contentEquals(other: ShortArray?): Boolean {\n   definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayEquals\")\npublic actual infix fun IntArray?.contentEquals(other: IntArray?): Boolean {\n   definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayEquals\")\npublic actual infix fun LongArray?.contentEquals(other: LongArray?): Boolean {\n   definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayEquals\")\npublic actual infix fun FloatArray?.contentEquals(other: FloatArray?): Boolean {\n   definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayEquals\")\npublic actual infix fun DoubleArray?.contentEquals(other: DoubleArray?): Boolean {\n   definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayEquals\")\npublic actual infix fun BooleanArray?.contentEquals(other: BooleanArray?): Boolean {\n   definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it

means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayEquals\")\npublic actual infix fun CharArray?.contentEquals(other: CharArray?): Boolean {\n definedExternally\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun <T> Array<out T>.contentHashCode(): Int {\n return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun ByteArray.contentHashCode(): Int {\n return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun ShortArray.contentHashCode(): Int {\n return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun IntArray.contentHashCode(): Int {\n return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun LongArray.contentHashCode(): Int {\n return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun FloatArray.contentHashCode(): Int {\n return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun DoubleArray.contentHashCode(): Int {\n return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun BooleanArray.contentHashCode(): Int {\n return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun CharArray.contentHashCode(): Int {\n return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayHashCode\")\npublic actual fun <T> Array<out T>?.contentHashCode(): Int {\n definedExternally\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayHashCode\")\npublic actual fun ByteArray?.contentHashCode(): Int {\n definedExternally\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayHashCode\")\npublic actual fun ShortArray?.contentHashCode(): Int {\n definedExternally\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayHashCode\")\npublic actual fun IntArray?.contentHashCode(): Int {\n definedExternally\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayHashCode\")\npublic actual fun LongArray?.contentHashCode(): Int {\n definedExternally\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayHashCode\")\npublic actual fun FloatArray?.contentHashCode(): Int {\n definedExternally\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayHashCode\")\npublic actual fun DoubleArray?.contentHashCode(): Int {\n definedExternally\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayHashCode\")\npublic actual fun BooleanArray?.contentHashCode(): Int {\n definedExternally\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayHashCode\")\npublic actual fun CharArray?.contentHashCode(): Int {\n definedExternally\n}\n\n/**\n * Returns a string representation of the

contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun <T> Array<out T>.contentToString(): String {\n   return this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun ByteArray.contentToString(): String {\n   return this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun ShortArray.contentToString(): String {\n   return this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun IntArray.contentToString(): String {\n   return this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun LongArray.contentToString(): String {\n   return this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun FloatArray.contentToString(): String {\n   return this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun DoubleArray.contentToString(): String {\n   return this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun BooleanArray.contentToString(): String {\n   return this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.1\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\npublic actual fun CharArray.contentToString(): String {\n   return this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayToString\")\npublic actual fun <T> Array<out T>?.contentToString(): String {\n   definedExternally\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayToString\")\npublic actual fun ByteArray?.contentToString(): String {\n   definedExternally\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayToString\")\npublic actual fun ShortArray?.contentToString(): String {\n   definedExternally\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayToString\")\npublic actual fun IntArray?.contentToString(): String {\n

definedExternally\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayToString\")\npublic actual fun LongArray?.contentToString(): String {\n   definedExternally\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayToString\")\npublic actual fun FloatArray?.contentToString(): String {\n   definedExternally\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayToString\")\npublic actual fun DoubleArray?.contentToString(): String {\n   definedExternally\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayToString\")\npublic actual fun BooleanArray?.contentToString(): String {\n   definedExternally\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin(\"1.4\")\n@library(\"arrayToString\")\npublic actual fun CharArray?.contentToString(): String {\n   definedExternally\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {\n   arrayCopy(this, destination, destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual inline fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray {\n   arrayCopy(this.unsafeCast<Array<Byte>>(), destination.unsafeCast<Array<Byte>>(), destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this

array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual inline fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray {\n   arrayCopy(this.unsafeCast<Array<Short>>(), destination.unsafeCast<Array<Short>>(), destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual inline fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray {\n   arrayCopy(this.unsafeCast<Array<Int>>(), destination.unsafeCast<Array<Int>>(), destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual inline fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): LongArray {\n   arrayCopy(this.unsafeCast<Array<Long>>(), destination.unsafeCast<Array<Long>>(), destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual inline fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): FloatArray {\n   arrayCopy(this.unsafeCast<Array<Float>>(), destination.unsafeCast<Array<Float>>(), destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n *

@param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual inline fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): DoubleArray {\n   arrayCopy(this.unsafeCast<Array<Double>>(), destination.unsafeCast<Array<Double>>(), destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual inline fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): BooleanArray {\n   arrayCopy(this.unsafeCast<Array<Boolean>>(), destination.unsafeCast<Array<Boolean>>(), destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual inline fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray {\n   arrayCopy(this.unsafeCast<Array<Char>>(), destination.unsafeCast<Array<Char>>(), destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\n@Suppress(\"ACTUAL_WITHOUT_EXPECT\", \"NOTHING_TO_INLINE\")\npublic actual inline fun <T> Array<out T>.copyOf(): Array<T> {\n   return this.asDynamic().slice()\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline fun ByteArray.copyOf(): ByteArray {\n   return this.asDynamic().slice()\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline fun ShortArray.copyOf(): ShortArray {\n   return this.asDynamic().slice()\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample

samples.collections.Arrays.CopyOfOperations.copyOf\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline fun IntArray.copyOf(): IntArray {\n   return this.asDynamic().slice()\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic actual fun LongArray.copyOf(): LongArray {\n   return withType(\"LongArray\", this.asDynamic().slice())\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline fun FloatArray.copyOf(): FloatArray {\n   return this.asDynamic().slice()\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline fun DoubleArray.copyOf(): DoubleArray {\n   return this.asDynamic().slice()\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic actual fun BooleanArray.copyOf(): BooleanArray {\n   return withType(\"BooleanArray\", this.asDynamic().slice())\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n */\npublic actual fun CharArray.copyOf(): CharArray {\n   return withType(\"CharArray\", this.asDynamic().slice())\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic actual fun ByteArray.copyOf(newSize: Int): ByteArray {\n   require(newSize >= 0) { \"Invalid new array size: $newSize.\" }\n   return fillFrom(this, ByteArray(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic actual fun ShortArray.copyOf(newSize: Int): ShortArray {\n   require(newSize >= 0) { \"Invalid new array size: $newSize.\" }\n   return fillFrom(this, ShortArray(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic actual fun IntArray.copyOf(newSize: Int): IntArray {\n   require(newSize >= 0) { \"Invalid new array size: $newSize.\" }\n   return fillFrom(this, IntArray(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic actual fun LongArray.copyOf(newSize: Int): LongArray {\n   require(newSize >= 0) { \"Invalid new array size: $newSize.\" }\n   return withType(\"LongArray\", arrayCopyResize(this, newSize, 0L))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic actual fun FloatArray.copyOf(newSize: Int): FloatArray {\n   require(newSize >= 0) { \"Invalid new array size: $newSize.\" }\n   return fillFrom(this, FloatArray(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original

array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic actual fun DoubleArray.copyOf(newSize: Int): DoubleArray {\n require(newSize >= 0) { \"Invalid new array size: $newSize.\" }\n return fillFrom(this, DoubleArray(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with `false` values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `false` values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic actual fun BooleanArray.copyOf(newSize: Int): BooleanArray {\n require(newSize >= 0) { \"Invalid new array size: $newSize.\" }\n return withType(\"BooleanArray\", arrayCopyResize(this, newSize, false))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with null char (`\\u0000`) values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with null char (`\\u0000`) values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic actual fun CharArray.copyOf(newSize: Int): CharArray {\n require(newSize >= 0) { \"Invalid new array size: $newSize.\" }\n return withType(\"CharArray\", fillFrom(this, CharArray(newSize)))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with `null` values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `null` values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizingCopyOf\n */\n@Suppress(\"ACTUAL_WITHOUT_EXPECT\")\npublic actual fun <T> Array<out T>.copyOf(newSize: Int): Array<T?> {\n require(newSize >= 0) { \"Invalid new array size: $newSize.\" }\n return arrayCopyResize(this, newSize, null)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@Suppress(\"ACTUAL_WITHOUT_EXPECT\")\npublic actual fun <T> Array<out T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T> {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n return this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic actual fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n return this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic actual fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n return this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to

copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic actual fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n return this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic actual fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n return withType(\"LongArray\", this.asDynamic().slice(fromIndex, toIndex))\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic actual fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n return this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic actual fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n return this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic actual fun BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n return withType(\"BooleanArray\", this.asDynamic().slice(fromIndex, toIndex))\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\npublic actual fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n return withType(\"CharArray\", this.asDynamic().slice(fromIndex, toIndex))\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun <T> Array<T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n this.asDynamic().fill(element, fromIndex, toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic

actual fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n this.asDynamic().fill(element, fromIndex, toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n this.asDynamic().fill(element, fromIndex, toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n this.asDynamic().fill(element, fromIndex, toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n this.asDynamic().fill(element, fromIndex, toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n this.asDynamic().fill(element, fromIndex, toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n this.asDynamic().fill(element, fromIndex, toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic

actual fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n this.asDynamic().fill(element, fromIndex, toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n this.asDynamic().fill(element, fromIndex, toIndex);\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"ACTUAL_WITHOUT_EXPECT\", \"NOTHING_TO_INLINE\")\npublic actual inline operator fun <T> Array<out T>.plus(element: T): Array<T> {\n return this.asDynamic().concat(arrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun ByteArray.plus(element: Byte): ByteArray {\n return plus(byteArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun ShortArray.plus(element: Short): ShortArray {\n return plus(shortArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun IntArray.plus(element: Int): IntArray {\n return plus(intArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun LongArray.plus(element: Long): LongArray {\n return plus(longArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun FloatArray.plus(element: Float): FloatArray {\n return plus(floatArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun DoubleArray.plus(element: Double): DoubleArray {\n return plus(doubleArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun BooleanArray.plus(element: Boolean): BooleanArray {\n return plus(booleanArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun CharArray.plus(element: Char): CharArray {\n return plus(charArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n@Suppress(\"ACTUAL_WITHOUT_EXPECT\")\npublic actual operator fun <T> Array<out T>.plus(elements: Collection<T>): Array<T> {\n return arrayPlusCollection(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic actual operator fun ByteArray.plus(elements: Collection<Byte>): ByteArray {\n return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic actual operator fun ShortArray.plus(elements: Collection<Short>): ShortArray {\n return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic actual operator fun IntArray.plus(elements: Collection<Int>): IntArray {\n return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic actual operator fun LongArray.plus(elements: Collection<Long>): LongArray {\n return arrayPlusCollection(this, elements)\n}\n\n/**\n * Returns an array

containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic actual operator fun FloatArray.plus(elements: Collection<Float>): FloatArray {\n   return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic actual operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray {\n   return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic actual operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray {\n   return arrayPlusCollection(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic actual operator fun CharArray.plus(elements: Collection<Char>): CharArray {\n   return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@Suppress(\"ACTUAL_WITHOUT_EXPECT\", \"NOTHING_TO_INLINE\")\npublic actual inline operator fun <T> Array<out T>.plus(elements: Array<out T>): Array<T> {\n   return this.asDynamic().concat(elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun ByteArray.plus(elements: ByteArray): ByteArray {\n   return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun ShortArray.plus(elements: ShortArray): ShortArray {\n   return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun IntArray.plus(elements: IntArray): IntArray {\n   return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun LongArray.plus(elements: LongArray): LongArray {\n   return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun FloatArray.plus(elements: FloatArray): FloatArray {\n   return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray {\n   return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray {\n   return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline operator fun CharArray.plus(elements: CharArray): CharArray {\n   return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@Suppress(\"ACTUAL_WITHOUT_EXPECT\", \"NOTHING_TO_INLINE\")\npublic actual inline fun <T> Array<out T>.plusElement(element: T): Array<T> {\n   return this.asDynamic().concat(arrayOf(element))\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\n@library(\"primitiveArraySort\")\npublic actual fun IntArray.sort(): Unit {\n   definedExternally\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic actual fun LongArray.sort(): Unit {\n   @Suppress(\"DEPRECATION\")\n   if (size > 1) sort { a: Long, b: Long -> a.compareTo(b) }\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n

*/\n@library(\"primitiveArraySort\")\npublic actual fun ByteArray.sort(): Unit {\n definedExternally\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\n@library(\"primitiveArraySort\")\npublic actual fun ShortArray.sort(): Unit {\n definedExternally\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\n@library(\"primitiveArraySort\")\npublic actual fun DoubleArray.sort(): Unit {\n definedExternally\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\n@library(\"primitiveArraySort\")\npublic actual fun FloatArray.sort(): Unit {\n definedExternally\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\n@library(\"primitiveArraySort\")\npublic actual fun CharArray.sort(): Unit {\n definedExternally\n}\n\n/**\n * Sorts the array in-place according to the natural order of its elements.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n * @sample samples.collections.Arrays.Sorting.sortArrayOfComparable\n */\npublic actual fun <T : Comparable<T>> Array<out T>.sort(): Unit {\n if (size > 1) sortArray(this)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given [comparison] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\n@Deprecated(\"Use sortWith instead\", ReplaceWith(\"this.sortWith(Comparator(comparison))\"))\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\npublic fun <T> Array<out T>.sort(comparison: (a: T, b: T) -> Int): Unit {\n if (size > 1) sortArrayWith(this, comparison)\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArrayOfComparable\n */\n@SinceKotlin(\"1.4\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun <T : Comparable<T>> Array<out T>.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n sortArrayWith(this, fromIndex, toIndex, naturalOrder())\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun ByteArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<ByteArray>()\n subarray.sort()\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun ShortArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<ShortArray>()\n subarray.sort()\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample

samples.collections.Arrays.Sorting.sortRangeOfArray\n
*/\n@SinceKotlin(\"1.4\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun IntArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<IntArray>()\n   subarray.sort()\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun LongArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   sortArrayWith(this.unsafeCast<Array<Long>>(), fromIndex, toIndex, naturalOrder())\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun FloatArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<FloatArray>()\n   subarray.sort()\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun DoubleArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<DoubleArray>()\n   subarray.sort()\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun CharArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<CharArray>()\n   subarray.sort()\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given [comparison] function.\n */\n@Deprecated(\"Use other sorting functions from the Standard Library\")\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.sort(noinline comparison: (a: Byte, b: Byte) -> Int): Unit {\n   asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given [comparison] function.\n */\n@Deprecated(\"Use other sorting functions from the Standard Library\")\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.sort(noinline comparison: (a: Short, b: Short) -> Int): Unit {\n asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given

[comparison] function.\n */\n@Deprecated(\"Use other sorting functions from the Standard Library\")\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.sort(noinline comparison: (a: Int, b: Int) -> Int): Unit {\n   asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given [comparison] function.\n */\n@Deprecated(\"Use other sorting functions from the Standard Library\")\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.sort(noinline comparison: (a: Long, b: Long) -> Int): Unit {\n asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given [comparison] function.\n */\n@Deprecated(\"Use other sorting functions from the Standard Library\")\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.sort(noinline comparison: (a: Float, b: Float) -> Int): Unit {\n asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given [comparison] function.\n */\n@Deprecated(\"Use other sorting functions from the Standard Library\")\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.sort(noinline comparison: (a: Double, b: Double) -> Int): Unit {\n asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given [comparison] function.\n */\n@Deprecated(\"Use other sorting functions from the Standard Library\")\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.sort(noinline comparison: (a: Char, b: Char) -> Int): Unit {\n asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given [comparator].\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic actual fun <T> Array<out T>.sortWith(comparator: Comparator<in T>): Unit {\n   if (size > 1) sortArrayWith(this, comparator)\n}\n\n/**\n * Sorts a range in the array in-place with the given [comparator].\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun <T> Array<out T>.sortWith(comparator: Comparator<in T>, fromIndex: Int = 0, toIndex: Int = size): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   sortArrayWith(this, fromIndex, toIndex, comparator)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic actual fun ByteArray.toTypedArray(): Array<Byte> {\n   return js(\"[]\").slice.call(this)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic actual fun ShortArray.toTypedArray(): Array<Short> {\n   return js(\"[]\").slice.call(this)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic actual fun IntArray.toTypedArray(): Array<Int> {\n   return js(\"[]\").slice.call(this)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic actual fun LongArray.toTypedArray(): Array<Long> {\n   return js(\"[]\").slice.call(this)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic actual fun FloatArray.toTypedArray(): Array<Float> {\n   return js(\"[]\").slice.call(this)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic actual fun DoubleArray.toTypedArray(): Array<Double> {\n   return js(\"[]\").slice.call(this)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic actual fun BooleanArray.toTypedArray(): Array<Boolean> {\n   return js(\"[]\").slice.call(this)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic actual fun CharArray.toTypedArray(): Array<Char> {\n   return Array(size) { index -> this[index] }\n}\n\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the

license/LICENSE.txt file.\n
*/\n@file:kotlin.jvm.JvmName(\"ComparisonsKt\")\n@file:kotlin.jvm.JvmMultifileClass\n\npackage kotlin.comparisons\n\n/**\n * Compares two values using the specified functions [selectors] to calculate the result of the comparison.\n * The functions are called sequentially, receive the given values [a] and [b] and return [Comparable]\n * objects. As soon as the [Comparable] instances returned by a function for [a] and [b] values do not\n * compare as equal, the result of that comparison is returned.\n *\n * @sample samples.comparisons.Comparisons.compareValuesByWithSelectors\n */\npublic fun <T> compareValuesBy(a: T, b: T, vararg selectors: (T) -> Comparable<*>?): Int {\n   require(selectors.size > 0)\n   return compareValuesByImpl(a, b, selectors)\n}\n\nprivate fun <T> compareValuesByImpl(a: T, b: T, selectors: Array<out (T) -> Comparable<*>?>): Int {\n   for (fn in selectors) {\n   val v1 = fn(a)\n   val v2 = fn(b)\n   val diff = compareValues(v1, v2)\n   if (diff != 0) return diff\n   }\n   return 0\n}\n\n/**\n * Compares two values using the specified [selector] function to calculate the result of the comparison.\n * The function is applied to the given values [a] and [b] and return [Comparable] objects.\n * The result of comparison of these [Comparable] instances is returned.\n *\n * @sample samples.comparisons.Comparisons.compareValuesByWithSingleSelector\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> compareValuesBy(a: T, b: T, selector: (T) -> Comparable<*>?): Int {\n   return compareValues(selector(a), selector(b))\n}\n\n/**\n * Compares two values using the specified [selector] function to calculate the result of the comparison.\n * The function is applied to the given values [a] and [b] and return objects of type K which are then being\n * compared with the given [comparator].\n *\n * @sample samples.comparisons.Comparisons.compareValuesByWithComparator\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T, K> compareValuesBy(a: T, b: T, comparator: Comparator<in K>, selector: (T) -> K): Int {\n   return comparator.compare(selector(a), selector(b))\n}\n\n//// Not so useful without type inference for receiver of expression\n//// compareValuesWith(v1, v2, compareBy { it.prop1 } thenByDescending { it.prop2 })\n///**\n// * Compares two values using the specified [comparator].\n// */\n//@Suppress(\"NOTHING_TO_INLINE\")\n//public inline fun <T> compareValuesWith(a: T, b: T, comparator: Comparator<T>): Int = comparator.compare(a, b)\n//\n\n/**\n * Compares two nullable [Comparable] values. Null is considered less than any value.\n *\n * @sample samples.comparisons.Comparisons.compareValues\n */\npublic fun <T : Comparable<*>> compareValues(a: T?, b: T?): Int {\n   if (a === b) return 0\n   if (a == null) return -1\n   if (b == null) return 1\n\n   @Suppress(\"UNCHECKED_CAST\")\n   return (a as Comparable<Any>).compareTo(b)\n}\n\n/**\n * Creates a comparator using the sequence of functions to calculate a result of comparison.\n * The functions are called sequentially, receive the given values `a` and `b` and return [Comparable]\n * objects. As soon as the [Comparable] instances returned by a function for `a` and `b` values do not\n * compare as equal, the result of that comparison is returned from the [Comparator].\n *\n * @sample samples.comparisons.Comparisons.compareByWithSelectors\n */\npublic fun <T> compareBy(vararg selectors: (T) -> Comparable<*>?): Comparator<T> {\n   require(selectors.size > 0)\n   return Comparator { a, b -> compareValuesByImpl(a, b, selectors) }\n}\n\n/**\n * Creates a comparator using the function to transform value to a [Comparable] instance for comparison.\n *\n * @sample samples.comparisons.Comparisons.compareByWithSingleSelector\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> compareBy(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =\n   Comparator { a, b -> compareValuesBy(a, b, selector) }\n\n/**\n * Creates a comparator using the [selector] function to transform values being compared and then applying\n * the specified [comparator] to compare transformed values.\n *\n * @sample samples.comparisons.Comparisons.compareByWithComparator\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T, K> compareBy(comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =\n   Comparator { a, b -> compareValuesBy(a, b, comparator, selector) }\n\n/**\n * Creates a descending comparator using the function to transform value to a [Comparable] instance for comparison.\n *\n * @sample samples.comparisons.Comparisons.compareByDescendingWithSingleSelector\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> compareByDescending(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =\n   Comparator { a, b -> compareValuesBy(b, a, selector) }\n\n/**\n * Creates a descending comparator using the [selector] function to transform values being compared and then

applying\n * the specified [comparator] to compare transformed values.\n *\n * Note that an order of [comparator] is reversed by this wrapper.\n *\n * @sample samples.comparisons.Comparisons.compareByDescendingWithComparator\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T, K> compareByDescending(comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =\n   Comparator { a, b -> compareValuesBy(b, a, comparator, selector) }\n\n/**\n * Creates a comparator comparing values after the primary comparator defined them equal. It uses\n * the function to transform value to a [Comparable] instance for comparison.\n *\n * @sample samples.comparisons.Comparisons.thenBy\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Comparator<T>.thenBy(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =\n   Comparator { a, b ->\n   val previousCompare = this@thenBy.compare(a, b)\n   if (previousCompare != 0) previousCompare else compareValuesBy(a, b, selector)\n   }\n\n/**\n * Creates a comparator comparing values after the primary comparator defined them equal. It uses\n * the [selector] function to transform values and then compares them with the given [comparator].\n *\n * @sample samples.comparisons.Comparisons.thenByWithComparator\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T, K> Comparator<T>.thenBy(comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =\n   Comparator { a, b ->\n   val previousCompare = this@thenBy.compare(a, b)\n   if (previousCompare != 0) previousCompare else compareValuesBy(a, b, comparator, selector)\n   }\n\n/**\n * Creates a descending comparator using the primary comparator and\n * the function to transform value to a [Comparable] instance for comparison.\n *\n * @sample samples.comparisons.Comparisons.thenByDescending\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Comparator<T>.thenByDescending(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =\n   Comparator { a, b ->\n   val previousCompare = this@thenByDescending.compare(a, b)\n   if (previousCompare != 0) previousCompare else compareValuesBy(b, a, selector)\n   }\n\n/**\n * Creates a descending comparator comparing values after the primary comparator defined them equal. It uses\n * the [selector] function to transform values and then compares them with the given [comparator].\n *\n * @sample samples.comparisons.Comparisons.thenByDescendingWithComparator\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T, K> Comparator<T>.thenByDescending(comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =\n   Comparator { a, b ->\n   val previousCompare = this@thenByDescending.compare(a, b)\n   if (previousCompare != 0) previousCompare else compareValuesBy(b, a, comparator, selector)\n   }\n\n\n/**\n * Creates a comparator using the primary comparator and function to calculate a result of comparison.\n *\n * @sample samples.comparisons.Comparisons.thenComparator\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Comparator<T>.thenComparator(crossinline comparison: (a: T, b: T) -> Int): Comparator<T> =\n   Comparator { a, b ->\n   val previousCompare = this@thenComparator.compare(a, b)\n   if (previousCompare != 0) previousCompare else comparison(a, b)\n   }\n\n/**\n * Combines this comparator and the given [comparator] such that the latter is applied only\n * when the former considered values equal.\n *\n * @sample samples.comparisons.Comparisons.then\n */\npublic infix fun <T> Comparator<T>.then(comparator: Comparator<in T>): Comparator<T> =\n   Comparator { a, b ->\n   val previousCompare = this@then.compare(a, b)\n   if (previousCompare != 0) previousCompare else comparator.compare(a, b)\n   }\n\n/**\n * Combines this comparator and the given [comparator] such that the latter is applied only\n * when the former considered values equal.\n *\n * @sample samples.comparisons.Comparisons.thenDescending\n */\npublic infix fun <T> Comparator<T>.thenDescending(comparator: Comparator<in T>): Comparator<T> =\n   Comparator<T> { a, b ->\n   val previousCompare = this@thenDescending.compare(a, b)\n   if (previousCompare != 0) previousCompare else comparator.compare(b, a)\n   }\n\n// Not so useful without type inference for receiver of expression\n/**\n * Extends the given [comparator] of non-nullable values to a comparator of nullable values\n * considering `null` value less than any other value.\n *\n * @sample samples.comparisons.Comparisons.nullsFirstLastWithComparator\n */\npublic fun <T : Any> nullsFirst(comparator: Comparator<in T>): Comparator<T?> =\n   Comparator { a, b ->\n   when {\n   a === b -> 0\n   a == null -> -1\n   b == null -> 1\n   else -> comparator.compare(a, b)\n   }\n   }\n\n/**\n * Provides a comparator of nullable [Comparable] values\n * considering `null` value less than any other

value.\n *\n * @sample samples.comparisons.Comparisons.nullsFirstLastComparator\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T : Comparable<T>> nullsFirst(): Comparator<T?> = nullsFirst(naturalOrder())\n\n/**\n * Extends the given [comparator] of non-nullable values to a comparator of nullable values\n * considering `null` value greater than any other value.\n *\n * @sample samples.comparisons.Comparisons.nullsFirstLastWithComparator\n */\npublic fun <T : Any> nullsLast(comparator: Comparator<in T>): Comparator<T?> =\n   Comparator { a, b ->\n   when {\n   a === b -> 0\n   a == null -> 1\n   b == null -> -1\n   else -> comparator.compare(a, b)\n   }\n   }\n\n/**\n * Provides a comparator of nullable [Comparable] values\n * considering `null` value greater than any other value.\n *\n * @sample samples.comparisons.Comparisons.nullsFirstLastComparator\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T : Comparable<T>> nullsLast(): Comparator<T?> = nullsLast(naturalOrder())\n\n/**\n * Returns a comparator that compares [Comparable] objects in natural order.\n *\n * @sample samples.comparisons.Comparisons.naturalOrderComparator\n */\npublic fun <T : Comparable<T>> naturalOrder(): Comparator<T> = @Suppress(\"UNCHECKED_CAST\") (NaturalOrderComparator as Comparator<T>)\n\n/**\n * Returns a comparator that compares [Comparable] objects in reversed natural order.\n *\n * @sample samples.comparisons.Comparisons.nullsFirstLastWithComparator\n */\npublic fun <T : Comparable<T>> reverseOrder(): Comparator<T> = @Suppress(\"UNCHECKED_CAST\") (ReverseOrderComparator as Comparator<T>)\n\n/**\n * Returns a comparator that imposes the reverse ordering of this comparator.\n *\n * @sample samples.comparisons.Comparisons.reversed\n */\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\")\npublic fun <T> Comparator<T>.reversed(): Comparator<T> = when (this) {\n   is ReversedComparator -> this.comparator\n   NaturalOrderComparator -> @Suppress(\"UNCHECKED_CAST\") (ReverseOrderComparator as Comparator<T>)\n   ReverseOrderComparator -> @Suppress(\"UNCHECKED_CAST\") (NaturalOrderComparator as Comparator<T>)\n   else -> ReversedComparator(this)\n}\n\n\nprivate class ReversedComparator<T>(public val comparator: Comparator<T>) : Comparator<T> {\n   override fun compare(a: T, b: T): Int = comparator.compare(b, a)\n   @Suppress(\"VIRTUAL_MEMBER_HIDDEN\")\n   fun reversed(): Comparator<T> = comparator\n}\n\nprivate object NaturalOrderComparator : Comparator<Comparable<Any>> {\n   override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int = a.compareTo(b)\n   @Suppress(\"VIRTUAL_MEMBER_HIDDEN\")\n   fun reversed(): Comparator<Comparable<Any>> = ReverseOrderComparator\n}\n\nprivate object ReverseOrderComparator : Comparator<Comparable<Any>> {\n   override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int = b.compareTo(a)\n   @Suppress(\"VIRTUAL_MEMBER_HIDDEN\")\n   fun reversed(): Comparator<Comparable<Any>> = NaturalOrderComparator\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"StandardKt\")\npackage kotlin\n\nimport kotlin.contracts.*\n\n/**\n * An exception is thrown to indicate that a method body remains to be implemented.\n */\npublic class NotImplementedError(message: String = \"An operation is not implemented.\") : Error(message)\n\n/**\n * Always throws [NotImplementedError] stating that operation is not implemented.\n */\n@kotlin.internal.InlineOnly\npublic inline fun TODO(): Nothing = throw NotImplementedError()\n\n/**\n * Always throws [NotImplementedError] stating that operation is not implemented.\n *\n * @param reason a string explaining why the implementation is missing.\n */\n@kotlin.internal.InlineOnly\npublic inline fun TODO(reason: String): Nothing = throw NotImplementedError(\"An operation is not implemented: $reason\")\n\n\n\n/**\n * Calls the specified function [block] and returns its result.\n *\n * For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#run).\n */\n@kotlin.internal.InlineOnly\npublic inline fun <R> run(block: () -> R): R {\n   contract {\n   callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n   }\n   return block()\n}\n\n/**\n * Calls the specified function [block] with `this` value as its receiver and returns its result.\n *\n * For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#run).\n

*/\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> T.run(block: T.() -> R): R {\n contract {\n callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n }\n return block()\n}\n\n/**\n * Calls the specified function [block] with the given [receiver] as its receiver and returns its result.\n *\n * For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#with).\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> with(receiver: T, block: T.() -> R): R {\n contract {\n callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n }\n return receiver.block()\n}\n\n/**\n * Calls the specified function [block] with `this` value as its receiver and returns `this` value.\n *\n * For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#apply).\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> T.apply(block: T.() -> Unit): T {\n contract {\n callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n }\n block()\n return this\n}\n\n/**\n * Calls the specified function [block] with `this` value as its argument and returns `this` value.\n *\n * For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#also).\n */\n@kotlin.internal.InlineOnly\n@SinceKotlin(\"1.1\")\npublic inline fun <T> T.also(block: (T) -> Unit): T {\n contract {\n callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n }\n block(this)\n return this\n}\n\n/**\n * Calls the specified function [block] with `this` value as its argument and returns its result.\n *\n * For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#let).\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> T.let(block: (T) -> R): R {\n contract {\n callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n }\n return block(this)\n}\n\n/**\n * Returns `this` value if it satisfies the given [predicate] or `null`, if it doesn't.\n *\n * For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#takeif-and-takeunless).\n */\n@kotlin.internal.InlineOnly\n@SinceKotlin(\"1.1\")\npublic inline fun <T> T.takeIf(predicate: (T) -> Boolean): T? {\n contract {\n callsInPlace(predicate, InvocationKind.EXACTLY_ONCE)\n }\n return if (predicate(this)) this else null\n}\n\n/**\n * Returns `this` value if it _does not_ satisfy the given [predicate] or `null`, if it does.\n *\n * For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#takeif-and-takeunless).\n */\n@kotlin.internal.InlineOnly\n@SinceKotlin(\"1.1\")\npublic inline fun <T> T.takeUnless(predicate: (T) -> Boolean): T? {\n contract {\n callsInPlace(predicate, InvocationKind.EXACTLY_ONCE)\n }\n return if (!predicate(this)) this else null\n}\n\n/**\n * Executes the given function [action] specified number of [times].\n *\n * A zero-based index of current iteration is passed as a parameter to [action].\n *\n * @sample samples.misc.ControlFlow.repeat\n */\n@kotlin.internal.InlineOnly\npublic inline fun repeat(times: Int, action: (Int) -> Unit) {\n contract { callsInPlace(action) }\n\n for (index in 0 until times) {\n action(index)\n }\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.comparisons\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.js.*\n\n/**\n * Returns the greater of two values.\n * \n * If values are equal, returns the first one.\n */\n@SinceKotlin(\"1.1\")\npublic actual fun <T : Comparable<T>> maxOf(a: T, b: T): T {\n return if (a >= b) a else b\n}\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Byte, b: Byte): Byte {\n return maxOf(a.toInt(), b.toInt()).unsafeCast<Byte>()\n}\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Short, b: Short): Short {\n return maxOf(a.toInt(), b.toInt()).unsafeCast<Short>()\n}\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Int, b: Int): Int {\n return JsMath.max(a, b)\n}\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.1\")\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline fun maxOf(a: Long, b: Long): Long {\n return if (a >= b) a else b\n}\n\n/**\n * Returns the greater of two values.\n * \n * If either value

is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Float, b: Float): Float {\n   return JsMath.max(a, b)\n}\n\n/**\n * Returns the greater of two values.\n * \n * If either value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Double, b: Double): Double {\n   return JsMath.max(a, b)\n}\n\n/**\n * Returns the greater of three values.\n * \n * If there are multiple equal maximal values, returns the first of them.\n */\n@SinceKotlin(\"1.1\")\npublic actual fun <T : Comparable<T>> maxOf(a: T, b: T, c: T): T {\n   return maxOf(a, maxOf(b, c))\n}\n\n/**\n * Returns the greater of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Byte, b: Byte, c: Byte): Byte {\n   return JsMath.max(a.toInt(), b.toInt(), c.toInt()).unsafeCast<Byte>()\n}\n\n/**\n * Returns the greater of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Short, b: Short, c: Short): Short {\n   return JsMath.max(a.toInt(), b.toInt(), c.toInt()).unsafeCast<Short>()\n}\n\n/**\n * Returns the greater of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Int, b: Int, c: Int): Int {\n   return JsMath.max(a, b, c)\n}\n\n/**\n * Returns the greater of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Long, b: Long, c: Long): Long {\n   return maxOf(a, maxOf(b, c))\n}\n\n/**\n * Returns the greater of three values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Float, b: Float, c: Float): Float {\n   return JsMath.max(a, b, c)\n}\n\n/**\n * Returns the greater of three values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Double, b: Double, c: Double): Double {\n   return JsMath.max(a, b, c)\n}\n\n/**\n * Returns the greater of the given values.\n * \n * If there are multiple equal maximal values, returns the first of them.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun <T : Comparable<T>> maxOf(a: T, vararg other: T): T {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun maxOf(a: Byte, vararg other: Byte): Byte {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun maxOf(a: Short, vararg other: Short): Short {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun maxOf(a: Int, vararg other: Int): Int {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun maxOf(a: Long, vararg other: Long): Long {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the greater of the given values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun maxOf(a: Float, vararg other: Float): Float {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the greater of the given values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun maxOf(a: Double, vararg other: Double): Double {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the smaller of two values.\n * \n * If values are equal, returns the first one.\n */\n@SinceKotlin(\"1.1\")\npublic actual fun <T : Comparable<T>> minOf(a: T, b: T): T {\n   return if (a <= b) a else b\n}\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Byte, b: Byte): Byte {\n   return minOf(a.toInt(), b.toInt()).unsafeCast<Byte>()\n}\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Short, b: Short): Short {\n   return minOf(a.toInt(), b.toInt()).unsafeCast<Short>()\n}\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Int, b: Int): Int {\n   return JsMath.min(a, b)\n}\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.1\")\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline fun minOf(a: Long, b: Long): Long {\n   return if (a <= b) a else b\n}\n\n/**\n * Returns the smaller of two values.\n * \n * If either value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Float, b: Float): Float {\n   return JsMath.min(a, b)\n}\n\n/**\n * Returns the smaller of two values.\n * \n * If either value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual

inline fun minOf(a: Double, b: Double): Double {\n   return JsMath.min(a, b)\n}\n\n/**\n * Returns the smaller of three values.\n * \n * If there are multiple equal minimal values, returns the first of them.\n */\n@SinceKotlin(\"1.1\")\npublic actual fun <T : Comparable<T>> minOf(a: T, b: T, c: T): T {\n   return minOf(a, minOf(b, c))\n}\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Byte, b: Byte, c: Byte): Byte {\n   return JsMath.min(a.toInt(), b.toInt(), c.toInt()).unsafeCast<Byte>()\n}\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Short, b: Short, c: Short): Short {\n   return JsMath.min(a.toInt(), b.toInt(), c.toInt()).unsafeCast<Short>()\n}\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Int, b: Int, c: Int): Int {\n   return JsMath.min(a, b, c)\n}\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Long, b: Long, c: Long): Long {\n   return minOf(a, minOf(b, c))\n}\n\n/**\n * Returns the smaller of three values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Float, b: Float, c: Float): Float {\n   return JsMath.min(a, b, c)\n}\n\n/**\n * Returns the smaller of three values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Double, b: Double, c: Double): Double {\n   return JsMath.min(a, b, c)\n}\n\n/**\n * Returns the smaller of the given values.\n * \n * If there are multiple equal minimal values, returns the first of them.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun <T : Comparable<T>> minOf(a: T, vararg other: T): T {\n   var min = a\n   for (e in other) min = minOf(min, e)\n   return min\n}\n\n/**\n * Returns the smaller of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun minOf(a: Byte, vararg other: Byte): Byte {\n   var min = a\n   for (e in other) min = minOf(min, e)\n   return min\n}\n\n/**\n * Returns the smaller of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun minOf(a: Short, vararg other: Short): Short {\n   var min = a\n   for (e in other) min = minOf(min, e)\n   return min\n}\n\n/**\n * Returns the smaller of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun minOf(a: Int, vararg other: Int): Int {\n   var min = a\n   for (e in other) min = minOf(min, e)\n   return min\n}\n\n/**\n * Returns the smaller of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun minOf(a: Long, vararg other: Long): Long {\n   var min = a\n   for (e in other) min = minOf(min, e)\n   return min\n}\n\n/**\n * Returns the smaller of the given values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun minOf(a: Float, vararg other: Float): Float {\n   var min = a\n   for (e in other) min = minOf(min, e)\n   return min\n}\n\n/**\n * Returns the smaller of the given values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun minOf(a: Double, vararg other: Double): Double {\n   var min = a\n   for (e in other) min = minOf(min, e)\n   return min\n}\n\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin\n\nimport kotlin.experimental.*\nimport kotlin.jvm.*\n\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@JvmInline\npublic value class ULong @PublishedApi internal constructor(@PublishedApi internal val data: Long) : Comparable<ULong> {\n\n   companion object {\n   /**\n   * A constant holding the minimum value an instance of ULong can have.\n   */\n   public const val MIN_VALUE: ULong = ULong(0)\n\n   /**\n   * A constant holding the maximum value an instance of ULong can have.\n   */\n   public const val MAX_VALUE: ULong = ULong(-1)\n\n   /**\n   * The number of bytes used to represent an instance of ULong in a binary form.\n   */\n   public const val SIZE_BYTES: Int = 8\n\n   /**\n   * The number of bits used to represent an instance of ULong in a binary form.\n   */\n   public const val SIZE_BITS: Int = 64\n   }\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toULong())\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a

positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toULong())\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun compareTo(other: UInt): Int = this.compareTo(other.toULong())\n\n   /**\n   * Compares this value with the specified value for order.\n   * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n   * or a positive number if it's greater than other.\n   */\n   @kotlin.internal.InlineOnly\n   @Suppress(\"OVERRIDE_BY_INLINE\")\n   public override inline operator fun compareTo(other: ULong): Int = ulongCompare(this.data, other.data)\n\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: UByte): ULong = this.plus(other.toULong())\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: UShort): ULong = this.plus(other.toULong())\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: UInt): ULong = this.plus(other.toULong())\n   /** Adds the other value to this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun plus(other: ULong): ULong = ULong(this.data.plus(other.data))\n\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: UByte): ULong = this.minus(other.toULong())\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: UShort): ULong = this.minus(other.toULong())\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: UInt): ULong = this.minus(other.toULong())\n   /** Subtracts the other value from this value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun minus(other: ULong): ULong = ULong(this.data.minus(other.data))\n\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: UByte): ULong = this.times(other.toULong())\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: UShort): ULong = this.times(other.toULong())\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: UInt): ULong = this.times(other.toULong())\n   /** Multiplies this value by the other value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun times(other: ULong): ULong = ULong(this.data.times(other.data))\n\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: UByte): ULong = this.div(other.toULong())\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: UShort): ULong = this.div(other.toULong())\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: UInt): ULong = this.div(other.toULong())\n   /** Divides this value by the other value, truncating the result to an integer that is closer to zero. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun div(other: ULong): ULong = ulongDivide(this, other)\n\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UByte): ULong = this.rem(other.toULong())\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UShort): ULong = this.rem(other.toULong())\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: UInt): ULong = this.rem(other.toULong())\n   /**\n   * Calculates the remainder of truncating division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rem(other: ULong): ULong = ulongRemainder(this, other)\n\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UByte): ULong =

this.floorDiv(other.toULong())\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UShort): ULong = this.floorDiv(other.toULong())\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: UInt): ULong = this.floorDiv(other.toULong())\n   /**\n   * Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.\n   * \n   * For unsigned types, the results of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun floorDiv(other: ULong): ULong = div(other)\n\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: UByte): UByte = this.mod(other.toULong()).toUByte()\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: UShort): UShort = this.mod(other.toULong()).toUShort()\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: UInt): UInt = this.mod(other.toULong()).toUInt()\n   /**\n   * Calculates the remainder of flooring division of this value by the other value.\n   * \n   * The result is always less than the divisor.\n   * \n   * For unsigned types, the remainders of flooring division and truncating division are the same.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun mod(other: ULong): ULong = rem(other)\n\n   /**\n   * Returns this value incremented by one.\n   *\n   * @sample samples.misc.Builtins.inc\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun inc(): ULong = ULong(data.inc())\n\n   /**\n   * Returns this value decremented by one.\n   *\n   * @sample samples.misc.Builtins.dec\n   */\n   @kotlin.internal.InlineOnly\n   public inline operator fun dec(): ULong = ULong(data.dec())\n\n   /** Creates a range from this value to the specified [other] value. */\n   @kotlin.internal.InlineOnly\n   public inline operator fun rangeTo(other: ULong): ULongRange = ULongRange(this, other)\n\n   /**\n   * Shifts this value left by the [bitCount] number of bits.\n   *\n   * Note that only the six lowest-order bits of the [bitCount] are used as the shift distance.\n   * The shift distance actually used is therefore always in the range `0..63`.\n   */\n   @kotlin.internal.InlineOnly\n   public inline infix fun shl(bitCount: Int): ULong = ULong(data shl bitCount)\n\n   /**\n   * Shifts this value right by the [bitCount] number of bits, filling the leftmost bits with zeros.\n   *\n   * Note that only the six lowest-order bits of the [bitCount] are used as the shift distance.\n   * The shift distance actually used is therefore always in the range `0..63`.\n   */\n   @kotlin.internal.InlineOnly\n   public inline infix fun shr(bitCount: Int): ULong = ULong(data ushr bitCount)\n\n   /** Performs a bitwise AND operation between the two values. */\n   @kotlin.internal.InlineOnly\n   public inline infix fun and(other: ULong): ULong = ULong(this.data and other.data)\n   /** Performs a bitwise OR operation between the two values. */\n   @kotlin.internal.InlineOnly\n   public inline infix fun or(other: ULong): ULong = ULong(this.data or other.data)\n   /** Performs a bitwise XOR operation between the two values. */\n   @kotlin.internal.InlineOnly\n   public inline infix fun xor(other: ULong): ULong = ULong(this.data xor other.data)\n   /** Inverts the bits in this value. */\n   @kotlin.internal.InlineOnly\n   public inline fun inv(): ULong = ULong(data.inv())\n\n   /**\n   * Converts this [ULong] value to [Byte].\n   *\n   * If this value is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents\n   * the same numerical value as this `ULong`.\n   *\n   * The resulting `Byte` value is represented by the least significant 8 bits of this `ULong` value.\n   * Note that the resulting `Byte` value may be negative.\n   */\n   @kotlin.internal.InlineOnly\n   public inline fun toByte(): Byte = data.toByte()\n   /**\n   * Converts this [ULong] value to [Short].\n   *\n   * If this value is less than or equals to [Short.MAX_VALUE], the resulting `Short` value represents\n   * the same numerical value as this `ULong`.\n   *\n   * The resulting `Short` value is represented

by the least significant 16 bits of this `ULong` value.\n * Note that the resulting `Short` value may be negative.\n */\n @kotlin.internal.InlineOnly\n public inline fun toShort(): Short = data.toShort()\n /**\n * Converts this [ULong] value to [Int].\n *\n * If this value is less than or equals to [Int.MAX_VALUE], the resulting `Int` value represents\n * the same numerical value as this `ULong`.\n *\n * The resulting `Int` value is represented by the least significant 32 bits of this `ULong` value.\n * Note that the resulting `Int` value may be negative.\n */\n @kotlin.internal.InlineOnly\n public inline fun toInt(): Int = data.toInt()\n /**\n * Converts this [ULong] value to [Long].\n *\n * If this value is less than or equals to [Long.MAX_VALUE], the resulting `Long` value represents\n * the same numerical value as this `ULong`. Otherwise the result is negative.\n *\n * The resulting `Long` value has the same binary representation as this `ULong` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toLong(): Long = data\n\n /**\n * Converts this [ULong] value to [UByte].\n *\n * If this value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents\n * the same numerical value as this `ULong`.\n *\n * The resulting `UByte` value is represented by the least significant 8 bits of this `ULong` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUByte(): UByte = data.toUByte()\n /**\n * Converts this [ULong] value to [UShort].\n *\n * If this value is less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents\n * the same numerical value as this `ULong`.\n *\n * The resulting `UShort` value is represented by the least significant 16 bits of this `ULong` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUShort(): UShort = data.toUShort()\n /**\n * Converts this [ULong] value to [UInt].\n *\n * If this value is less than or equals to [UInt.MAX_VALUE], the resulting `UInt` value represents\n * the same numerical value as this `ULong`.\n *\n * The resulting `UInt` value is represented by the least significant 32 bits of this `ULong` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUInt(): UInt = data.toUInt()\n /** Returns this value. */\n @kotlin.internal.InlineOnly\n public inline fun toULong(): ULong = this\n\n /**\n * Converts this [ULong] value to [Float].\n *\n * The resulting value is the closest `Float` to this `ULong` value.\n * In case when this `ULong` value is exactly between two `Float`s,\n * the one with zero at least significant bit of mantissa is selected.\n */\n @kotlin.internal.InlineOnly\n public inline fun toFloat(): Float = this.toDouble().toFloat()\n /**\n * Converts this [ULong] value to [Double].\n *\n * The resulting value is the closest `Double` to this `ULong` value.\n * In case when this `ULong` value is exactly between two `Double`s,\n * the one with zero at least significant bit of mantissa is selected.\n */\n @kotlin.internal.InlineOnly\n public inline fun toDouble(): Double = ulongToDouble(data)\n\n public override fun toString(): String = ulongToString(data)\n\n}\n\n/**\n * Converts this [Byte] value to [ULong].\n *\n * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Byte`.\n *\n * The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `Byte` value,\n * whereas the most significant 56 bits are filled with the sign bit of this value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Byte.toULong(): ULong = ULong(this.toLong())\n/**\n * Converts this [Short] value to [ULong].\n *\n * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Short`.\n *\n * The least significant 16 bits of the resulting `ULong` value are the same as the bits of this `Short` value,\n * whereas the most significant 48 bits are filled with the sign bit of this value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Short.toULong(): ULong = ULong(this.toLong())\n/**\n * Converts this [Int] value to [ULong].\n *\n * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Int`.\n *\n * The least significant 32 bits of the resulting `ULong` value are the same as the bits of this `Int` value,\n * whereas the most significant 32 bits are filled with the sign bit of this value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Int.toULong(): ULong = ULong(this.toLong())\n/**\n * Converts this [Long] value to [ULong].\n *\n * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Long`.\n *\n * The resulting `ULong` value has the same binary representation as this `Long` value.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Long.toULong(): ULong = ULong(this)\n\n/**\n * Converts this [Float] value to [ULong].\n *\n *

The fractional part, if any, is rounded down towards zero.\n * Returns zero if this `Float` value is negative or `NaN`, [ULong.MAX_VALUE] if it's bigger than `ULong.MAX_VALUE`.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Float.toULong(): ULong = doubleToULong(this.toDouble())\n/**\n * Converts this [Double] value to [ULong].\n *\n * The fractional part, if any, is rounded down towards zero.\n * Returns zero if this `Double` value is negative or `NaN`, [ULong.MAX_VALUE] if it's bigger than `ULong.MAX_VALUE`.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun Double.toULong(): ULong = doubleToULong(this)\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"CollectionsKt\")\n\npackage kotlin.collections\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\nimport kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns 1st *element* from the list.\n * \n * Throws an [IndexOutOfBoundsException] if the size of this list is less than 1.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> List<T>.component1(): T {\n   return get(0)\n}\n\n/**\n * Returns 2nd *element* from the list.\n * \n * Throws an [IndexOutOfBoundsException] if the size of this list is less than 2.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> List<T>.component2(): T {\n   return get(1)\n}\n\n/**\n * Returns 3rd *element* from the list.\n * \n * Throws an [IndexOutOfBoundsException] if the size of this list is less than 3.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> List<T>.component3(): T {\n   return get(2)\n}\n\n/**\n * Returns 4th *element* from the list.\n * \n * Throws an [IndexOutOfBoundsException] if the size of this list is less than 4.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> List<T>.component4(): T {\n   return get(3)\n}\n\n/**\n * Returns 5th *element* from the list.\n * \n * Throws an [IndexOutOfBoundsException] if the size of this list is less than 5.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> List<T>.component5(): T {\n   return get(4)\n}\n\n/**\n * Returns `true` if [element] is found in the collection.\n */\npublic operator fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.contains(element: T): Boolean {\n   if (this is Collection)\n   return contains(element)\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this collection.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic fun <T> Iterable<T>.elementAt(index: Int): T {\n   if (this is List)\n   return get(index)\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"Collection doesn't contain element at index $index.\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this list.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> List<T>.elementAt(index: Int): T {\n   return get(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this collection.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\npublic fun <T> Iterable<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {\n   if (this is List)\n   return this.getOrElse(index, defaultValue)\n   if (index < 0)\n   return defaultValue(index)\n   val iterator = iterator()\n   var count = 0\n   while (iterator.hasNext()) {\n   val element = iterator.next()\n   if (index == count++)\n   return element\n   }\n   return defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this list.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> List<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this collection.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\npublic fun <T>

```
Iterable<T>.elementAtOrNull(index: Int): T? {\n    if (this is List)\n        return this.getOrNull(index)\n    if (index < 0)\n        return null\n    val iterator = iterator()\n    var count = 0\n    while (iterator.hasNext()) {\n        val element = iterator.next()\n        if (index == count++)\n            return element\n    }\n    return null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this list.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> List<T>.elementAtOrNull(index: Int): T? {\n    return this.getOrNull(index)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.find(predicate: (T) -> Boolean): T? {\n    return firstOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.findLast(predicate: (T) -> Boolean): T? {\n    return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> List<T>.findLast(predicate: (T) -> Boolean): T? {\n    return lastOrNull(predicate)\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the collection is empty.\n */\npublic fun <T> Iterable<T>.first(): T {\n    when (this) {\n        is List -> return this.first()\n        else -> {\n            val iterator = iterator()\n            if (!iterator.hasNext())\n                throw NoSuchElementException(\"Collection is empty.\")\n            return iterator.next()\n        }\n    }\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the list is empty.\n */\npublic fun <T> List<T>.first(): T {\n    if (isEmpty())\n        throw NoSuchElementException(\"List is empty.\")\n    return this[0]\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun <T> Iterable<T>.first(predicate: (T) -> Boolean): T {\n    for (element in this) if (predicate(element)) return element\n    throw NoSuchElementException(\"Collection contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first non-null value produced by [transform] function being applied to elements of this collection in iteration order,\n * or throws [NoSuchElementException] if no non-null value was produced.\n * \n * @sample samples.collections.Collections.Transformations.firstNotNullOf\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Any> Iterable<T>.firstNotNullOf(transform: (T) -> R?): R {\n    return firstNotNullOfOrNull(transform) ?: throw NoSuchElementException(\"No element of the collection was transformed to a non-null value.\")\n}\n\n/**\n * Returns the first non-null value produced by [transform] function being applied to elements of this collection in iteration order,\n * or `null` if no non-null value was produced.\n * \n * @sample samples.collections.Collections.Transformations.firstNotNullOf\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Any> Iterable<T>.firstNotNullOfOrNull(transform: (T) -> R?): R? {\n    for (element in this) {\n        val result = transform(element)\n        if (result != null) {\n            return result\n        }\n    }\n    return null\n}\n\n/**\n * Returns the first element, or `null` if the collection is empty.\n */\npublic fun <T> Iterable<T>.firstOrNull(): T? {\n    when (this) {\n        is List -> {\n            if (isEmpty())\n                return null\n            else\n                return this[0]\n        }\n        else -> {\n            val iterator = iterator()\n            if (!iterator.hasNext())\n                return null\n            return iterator.next()\n        }\n    }\n}\n\n/**\n * Returns the first element, or `null` if the list is empty.\n */\npublic fun <T> List<T>.firstOrNull(): T? {\n    return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun <T> Iterable<T>.firstOrNull(predicate: (T) -> Boolean): T? {\n    for (element in this) if (predicate(element)) return element\n    return null\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this list.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> List<T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {\n    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this list.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n
```

```
*/\npublic fun <T> List<T>.getOrNull(index: Int): T? {\n    return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns first index of [element], or -1 if the collection does not contain element.\n */\npublic fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.indexOf(element: T): Int {\n    if (this is List) return this.indexOf(element)\n    var index = 0\n    for (item in this) {\n        checkIndexOverflow(index)\n        if (element == item)\n            return index\n        index++\n    }\n    return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the list does not contain element.\n */\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\") // false warning, extension takes precedence in some cases\npublic fun <@kotlin.internal.OnlyInputTypes T> List<T>.indexOf(element: T): Int {\n    return indexOf(element)\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the collection does not contain such element.\n */\npublic inline fun <T> Iterable<T>.indexOfFirst(predicate: (T) -> Boolean): Int {\n    var index = 0\n    for (item in this) {\n        checkIndexOverflow(index)\n        if (predicate(item))\n            return index\n        index++\n    }\n    return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the list does not contain such element.\n */\npublic inline fun <T> List<T>.indexOfFirst(predicate: (T) -> Boolean): Int {\n    var index = 0\n    for (item in this) {\n        if (predicate(item))\n            return index\n        index++\n    }\n    return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the collection does not contain such element.\n */\npublic inline fun <T> Iterable<T>.indexOfLast(predicate: (T) -> Boolean): Int {\n    var lastIndex = -1\n    var index = 0\n    for (item in this) {\n        checkIndexOverflow(index)\n        if (predicate(item))\n            lastIndex = index\n        index++\n    }\n    return lastIndex\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the list does not contain such element.\n */\npublic inline fun <T> List<T>.indexOfLast(predicate: (T) -> Boolean): Int {\n    val iterator = this.listIterator(size)\n    while (iterator.hasPrevious()) {\n        if (predicate(iterator.previous())) {\n            return iterator.nextIndex()\n        }\n    }\n    return -1\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the collection is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun <T> Iterable<T>.last(): T {\n    when (this) {\n        is List -> return this.last()\n        else -> {\n            val iterator = iterator()\n            if (!iterator.hasNext())\n                throw NoSuchElementException(\"Collection is empty.\")\n            var last = iterator.next()\n            while (iterator.hasNext())\n                last = iterator.next()\n            return last\n        }\n    }\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the list is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun <T> List<T>.last(): T {\n    if (isEmpty())\n        throw NoSuchElementException(\"List is empty.\")\n    return this[lastIndex]\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun <T> Iterable<T>.last(predicate: (T) -> Boolean): T {\n    var last: T? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            last = element\n            found = true\n        }\n    }\n    if (!found) throw NoSuchElementException(\"Collection contains no element matching the predicate.\")\n    @Suppress(\"UNCHECKED_CAST\")\n    return last as T\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun <T> List<T>.last(predicate: (T) -> Boolean): T {\n    val iterator = this.listIterator(size)\n    while (iterator.hasPrevious()) {\n        val element = iterator.previous()\n        if (predicate(element)) return element\n    }\n    throw NoSuchElementException(\"List contains no element matching the predicate.\")\n}\n\n/**\n * Returns last index of [element], or -1 if the collection does not contain element.\n */\npublic fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.lastIndexOf(element: T): Int {\n    if (this is List) return this.lastIndexOf(element)\n    var lastIndex = -1\n    var index = 0\n    for (item in this) {\n        checkIndexOverflow(index)\n        if (element == item)\n            lastIndex = index\n        index++\n    }\n    return lastIndex\n}\n\n/**\n * Returns last index of [element], or -1 if the list does not contain element.\n */\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\") // false warning, extension takes precedence in some cases\npublic fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {\n    return lastIndexOf(element)\n}\n\n/**\n * Returns the last element, or `null` if the collection is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun <T> Iterable<T>.lastOrNull(): T? {\n    when (this)
```

{\n is List -> return if (isEmpty()) null else this[size - 1]\n else -> {\n val iterator = iterator()\n if (!iterator.hasNext())\n return null\n var last = iterator.next()\n while (iterator.hasNext())\n last = iterator.next()\n return last\n }\n }\n}\n\n/**\n * Returns the last element, or `null` if the list is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun <T> List<T>.lastOrNull(): T? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun <T> Iterable<T>.lastOrNull(predicate: (T) -> Boolean): T? {\n var last: T? = null\n for (element in this) {\n if (predicate(element)) {\n last = element\n }\n }\n return last\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun <T> List<T>.lastOrNull(predicate: (T) -> Boolean): T? {\n val iterator = this.listIterator(size)\n while (iterator.hasPrevious()) {\n val element = iterator.previous()\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns a random element from this collection.\n * \n * @throws NoSuchElementException if this collection is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Collection<T>.random(): T {\n return random(Random)\n}\n\n/**\n * Returns a random element from this collection using the specified source of randomness.\n * \n * @throws NoSuchElementException if this collection is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun <T> Collection<T>.random(random: Random): T {\n if (isEmpty())\n throw NoSuchElementException(\"Collection is empty.\")\n return elementAt(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this collection, or `null` if this collection is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <T> Collection<T>.randomOrNull(): T? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this collection using the specified source of randomness, or `null` if this collection is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun <T> Collection<T>.randomOrNull(random: Random): T? {\n if (isEmpty())\n return null\n return elementAt(random.nextInt(size))\n}\n\n/**\n * Returns the single element, or throws an exception if the collection is empty or has more than one element.\n */\npublic fun <T> Iterable<T>.single(): T {\n when (this) {\n is List -> return this.single()\n else -> {\n val iterator = iterator()\n if (!iterator.hasNext())\n throw NoSuchElementException(\"Collection is empty.\")\n val single = iterator.next()\n if (iterator.hasNext())\n throw IllegalArgumentException(\"Collection has more than one element.\")\n return single\n }\n }\n}\n\n/**\n * Returns the single element, or throws an exception if the list is empty or has more than one element.\n */\npublic fun <T> List<T>.single(): T {\n return when (size) {\n 0 -> throw NoSuchElementException(\"List is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"List has more than one element.\")\n }\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun <T> Iterable<T>.single(predicate: (T) -> Boolean): T {\n var single: T? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Collection contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Collection contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as T\n}\n\n/**\n * Returns single element, or `null` if the collection is empty or has more than one element.\n */\npublic fun <T> Iterable<T>.singleOrNull(): T? {\n when (this) {\n is List -> return if (size == 1) this[0] else null\n else -> {\n val iterator = iterator()\n if (!iterator.hasNext())\n return null\n val single = iterator.next()\n if (iterator.hasNext())\n return null\n return single\n }\n }\n}\n\n/**\n * Returns single element, or `null` if the list is empty or has more than one element.\n */\npublic fun <T> List<T>.singleOrNull(): T? {\n return if (size == 1) this[0] else null\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic inline fun <T> Iterable<T>.singleOrNull(predicate: (T) -> Boolean): T? {\n var single: T? = null\n var found =

```
false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single = element\n            found = true\n        }\n    }\n    if (!found) return null\n    return single\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun <T> Iterable<T>.drop(n: Int): List<T> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return toList()\n    val list: ArrayList<T>\n    if (this is Collection<*>) {\n        val resultSize = size - n\n        if (resultSize <= 0)\n            return emptyList()\n        if (resultSize == 1)\n            return listOf(last())\n        list = ArrayList<T>(resultSize)\n        if (this is List<T>) {\n            if (this is RandomAccess) {\n                for (index in n until size)\n                    list.add(this[index])\n            } else {\n                for (item in listIterator(n))\n                    list.add(item)\n            }\n            return list\n        }\n    }\n    else {\n        list = ArrayList<T>()\n    }\n    var count = 0\n    for (item in this) {\n        if (count >= n) list.add(item) else ++count\n    }\n    return list.optimizeReadOnlyList()\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun <T> List<T>.dropLast(n: Int): List<T> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun <T> List<T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {\n    if (!isEmpty()) {\n        val iterator = listIterator(size)\n        while (iterator.hasPrevious()) {\n            if (!predicate(iterator.previous())) {\n                return take(iterator.nextIndex() + 1)\n            }\n        }\n    }\n    return emptyList()\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun <T> Iterable<T>.dropWhile(predicate: (T) -> Boolean): List<T> {\n    var yielding = false\n    val list = ArrayList<T>()\n    for (item in this)\n        if (yielding)\n            list.add(item)\n        else if (!predicate(item)) {\n            list.add(item)\n            yielding = true\n        }\n    return list\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun <T> Iterable<T>.filter(predicate: (T) -> Boolean): List<T> {\n    return filterTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun <T> Iterable<T>.filterIndexed(predicate: (index: Int, T) -> Boolean): List<T> {\n    return filterIndexedTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <T, C : MutableCollection<in T>> Iterable<T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {\n    forEachIndexed { index, element ->\n        if (predicate(index, element)) destination.add(element)\n    }\n    return destination\n}\n\n/**\n * Returns a list containing all elements that are instances of specified type parameter R.\n * \n * @sample samples.collections.Collections.Filtering.filterIsInstance\n */\npublic inline fun <reified R> Iterable<*>.filterIsInstance(): List<@kotlin.internal.NoInfer R> {\n    return filterIsInstanceTo(ArrayList<R>())\n}\n\n/**\n * Appends all elements that are instances of specified type parameter R to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterIsInstanceTo\n */\npublic inline fun <reified R, C : MutableCollection<in R>> Iterable<*>.filterIsInstanceTo(destination: C): C {\n    for (element in this) if (element is R) destination.add(element)\n    return destination\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun <T> Iterable<T>.filterNot(predicate: (T) -> Boolean): List<T> {\n    return filterNotTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Returns a list containing all elements that are not `null`.\n * \n * @sample samples.collections.Collections.Filtering.filterNotNull\n */\npublic fun <T : Any> Iterable<T?>.filterNotNull():
```

List<T> {\n return filterNotNullTo(ArrayList<T>())\n}\n\n/**\n * Appends all elements that are not `null` to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterNotNullTo\n */\npublic fun <C : MutableCollection<in T>, T : Any> Iterable<T?>.filterNotNullTo(destination: C): C {\n for (element in this) if (element != null) destination.add(element)\n return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <T, C : MutableCollection<in T>> Iterable<T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {\n for (element in this) if (!predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <T, C : MutableCollection<in T>> Iterable<T>.filterTo(destination: C, predicate: (T) -> Boolean): C {\n for (element in this) if (predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic fun <T> List<T>.slice(indices: IntRange): List<T> {\n if (indices.isEmpty()) return listOf()\n return this.subList(indices.start, indices.endInclusive + 1).toList()\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun <T> List<T>.slice(indices: Iterable<Int>): List<T> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<T>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun <T> Iterable<T>.take(n: Int): List<T> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n if (this is Collection<T>) {\n if (n >= size) return toList()\n if (n == 1) return listOf(first())\n }\n var count = 0\n val list = ArrayList<T>(n)\n for (item in this) {\n list.add(item)\n if (++count == n)\n break\n }\n return list.optimizeReadOnlyList()\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun <T> List<T>.takeLast(n: Int): List<T> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(last())\n val list = ArrayList<T>(n)\n if (this is RandomAccess) {\n for (index in size - n until size)\n list.add(this[index])\n } else {\n for (item in listIterator(size - n))\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun <T> List<T>.takeLastWhile(predicate: (T) -> Boolean): List<T> {\n if (isEmpty())\n return emptyList()\n val iterator = listIterator(size)\n while (iterator.hasPrevious()) {\n if (!predicate(iterator.previous())) {\n iterator.next()\n val expectedSize = size - iterator.nextIndex()\n if (expectedSize == 0) return emptyList()\n return ArrayList<T>(expectedSize).apply {\n while (iterator.hasNext())\n add(iterator.next())\n }\n }\n }\n return toList()\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic inline fun <T> Iterable<T>.takeWhile(predicate: (T) -> Boolean): List<T> {\n val list = ArrayList<T>()\n for (item in this) {\n if (!predicate(item))\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Reverses elements in the list in-place.\n */\npublic expect fun <T> MutableList<T>.reverse(): Unit\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun <T> Iterable<T>.reversed(): List<T> {\n if (this is Collection && size <= 1) return toList()\n val list = toMutableList()\n list.reverse()\n return list\n}\n\n/**\n * Randomly shuffles elements in this list in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.3\")\npublic fun <T> MutableList<T>.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1) {\n val j = random.nextInt(i + 1)\n this[j] = this.set(i, this[j])\n }\n}\n\n/**\n * Sorts elements in the list in-place according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic inline fun <T, R : Comparable<R>> MutableList<T>.sortBy(crossinline selector: (T) -> R?): Unit {\n if (size > 1)

sortWith(compareBy(selector))\n}\n\n/**\n * Sorts elements in the list in-place descending according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic inline fun <T, R : Comparable<R>> MutableList<T>.sortByDescending(crossinline selector: (T) -> R?): Unit {\n if (size > 1) sortWith(compareByDescending(selector))\n}\n\n/**\n * Sorts elements in the list in-place descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T : Comparable<T>> MutableList<T>.sortDescending(): Unit {\n sortWith(reverseOrder())\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T : Comparable<T>> Iterable<T>.sorted(): List<T> {\n if (this is Collection) {\n if (size <= 1) return this.toList()\n @Suppress(\"UNCHECKED_CAST\")\n return (toTypedArray<Comparable<T>>() as Array<T>).apply { sort() }.asList()\n }\n return toMutableList().apply { sort() }\n}\n\n/**\n * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <T, R : Comparable<R>> Iterable<T>.sortedBy(crossinline selector: (T) -> R?): List<T> {\n return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic inline fun <T, R : Comparable<R>> Iterable<T>.sortedByDescending(crossinline selector: (T) -> R?): List<T> {\n return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T : Comparable<T>> Iterable<T>.sortedDescending(): List<T> {\n return sortedWith(reverseOrder())\n}\n\n/**\n * Returns a list of all elements sorted according to the specified [comparator].\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic fun <T> Iterable<T>.sortedWith(comparator: Comparator<in T>): List<T> {\n if (this is Collection) {\n if (size <= 1) return this.toList()\n @Suppress(\"UNCHECKED_CAST\")\n return (toTypedArray<Any?>() as Array<T>).apply { sortWith(comparator) }.asList()\n }\n return toMutableList().apply { sortWith(comparator) }\n}\n\n/**\n * Returns an array of Boolean containing all of the elements of this collection.\n */\npublic fun Collection<Boolean>.toBooleanArray(): BooleanArray {\n val result = BooleanArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return result\n}\n\n/**\n * Returns an array of Byte containing all of the elements of this collection.\n */\npublic fun Collection<Byte>.toByteArray(): ByteArray {\n val result = ByteArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return result\n}\n\n/**\n * Returns an array of Char containing all of the elements of this collection.\n */\npublic fun Collection<Char>.toCharArray(): CharArray {\n val result = CharArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return result\n}\n\n/**\n * Returns an array of Double containing all of the elements of this collection.\n */\npublic fun Collection<Double>.toDoubleArray(): DoubleArray {\n val result = DoubleArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return result\n}\n\n/**\n * Returns an array of Float containing all of the elements of this collection.\n */\npublic fun Collection<Float>.toFloatArray(): FloatArray {\n val result = FloatArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return result\n}\n\n/**\n * Returns an array of Int containing all of the elements of this collection.\n */\npublic fun Collection<Int>.toIntArray(): IntArray {\n val result = IntArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return result\n}\n\n/**\n * Returns an array of Long containing all of the elements of this collection.\n */\npublic fun Collection<Long>.toLongArray(): LongArray {\n val result = LongArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return result\n}\n\n/**\n * Returns an array of Short containing all of the elements of this collection.\n */\npublic fun Collection<Short>.toShortArray(): ShortArray {\n val result = ShortArray(size)\n var index = 0\n for (element

in this)\n   result[index++] = element\n   return result\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given collection.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original collection.\n * \n * @sample samples.collections.Collections.Transformations.associate\n */\npublic inline fun <T, K, V> Iterable<T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {\n   val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing the elements from the given collection indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original collection.\n * \n * @sample samples.collections.Collections.Transformations.associateBy\n */\npublic inline fun <T, K> Iterable<T>.associateBy(keySelector: (T) -> K): Map<K, T> {\n   val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given collection.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original collection.\n * \n * @sample samples.collections.Collections.Transformations.associateByWithValueTransform\n */\npublic inline fun <T, K, V> Iterable<T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {\n   val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given collection\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Collections.Transformations.associateByTo\n */\npublic inline fun <T, K, M : MutableMap<in K, in T>> Iterable<T>.associateByTo(destination: M, keySelector: (T) -> K): M {\n   for (element in this) {\n   destination.put(keySelector(element), element)\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * value is provided by the [valueTransform] function applied to elements of the given collection.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.collections.Collections.Transformations.associateByToWithValueTransform\n */\npublic inline fun <T, K, V, M : MutableMap<in K, in V>> Iterable<T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {\n   for (element in this) {\n   destination.put(keySelector(element), valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given collection.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * @sample samples.collections.Collections.Transformations.associateTo\n */\npublic inline fun <T, K, V, M : MutableMap<in K, in V>> Iterable<T>.associateTo(destination: M, transform: (T) -> Pair<K, V>): M {\n   for (element in this) {\n   destination += transform(element)\n   }\n   return destination\n}\n\n/**\n * Returns a [Map] where keys are elements from the given collection and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original collection.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.3\")\npublic inline fun <K, V> Iterable<K>.associateWith(valueSelector: (K) -> V): Map<K, V> {\n   val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given collection,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the

map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.3\")\npublic inline fun <K, V, M : MutableMap<in K, in V>> Iterable<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n */\npublic fun <T, C : MutableCollection<in T>> Iterable<T>.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n */\npublic fun <T> Iterable<T>.toHashSet(): HashSet<T> {\n return toCollection(HashSet<T>(mapCapacity(collectionSizeOrDefault(12))))\n}\n\n/**\n * Returns a [List] containing all elements.\n */\npublic fun <T> Iterable<T>.toList(): List<T> {\n if (this is Collection) {\n return when (size) {\n 0 -> emptyList()\n 1 -> listOf(if (this is List) get(0) else iterator().next())\n else -> this.toMutableList()\n }\n }\n return this.toMutableList().optimizeReadOnlyList()\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this collection.\n */\npublic fun <T> Iterable<T>.toMutableList(): MutableList<T> {\n if (this is Collection<T>)\n return this.toMutableList()\n return toCollection(ArrayList<T>())\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this collection.\n */\npublic fun <T> Collection<T>.toMutableList(): MutableList<T> {\n return ArrayList(this)\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original collection.\n */\npublic fun <T> Iterable<T>.toSet(): Set<T> {\n if (this is Collection) {\n return when (size) {\n 0 -> emptySet()\n 1 -> setOf(if (this is List) this[0] else iterator().next())\n else -> toCollection(LinkedHashSet<T>(mapCapacity(size)))\n }\n }\n return toCollection(LinkedHashSet<T>()).optimizeReadOnlySet()\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original collection.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <T, R> Iterable<T>.flatMap(transform: (T) -> Iterable<R>): List<R> {\n return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original collection.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapSequence\")\npublic inline fun <T, R> Iterable<T>.flatMap(transform: (T) -> Sequence<R>): List<R> {\n return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original collection.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Iterable<T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): List<R> {\n return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original collection.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedSequence\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Iterable<T>.flatMapIndexed(transform: (index: Int, T) -> Sequence<R>): List<R> {\n return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original collection, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list =

transform(checkIndexOverflow(index++), element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original collection, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedSequenceTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Sequence<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(checkIndexOverflow(index++), element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original collection, to the given [destination].\n */\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original collection, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapSequenceTo\")\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Groups elements of the original collection by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original collection.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <T, K> Iterable<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {\n return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original collection\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original collection.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <T, K, V> Iterable<T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {\n return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups elements of the original collection by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<T>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original collection\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Creates a [Grouping] source from a collection to be used later with one of group-and-fold operations\n * using the specified [keySelector] function to extract a key from each element.\n * \n * @sample samples.collections.Grouping.groupingByEachCount\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K> Iterable<T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {\n return object : Grouping<T, K>

{\n override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()\n override fun keyOf(element: T): K = keySelector(element)\n }\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original collection.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic inline fun <T, R> Iterable<T>.map(transform: (T) -> R): List<R> {\n return mapTo(ArrayList<R>(collectionSizeOrDefault(10)), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original collection.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <T, R> Iterable<T>.mapIndexed(transform: (index: Int, T) -> R): List<R> {\n return mapIndexedTo(ArrayList<R>(collectionSizeOrDefault(10)), transform)\n}\n\n/**\n * Returns a list containing only the non-null results of applying the given [transform] function\n * to each element and its index in the original collection.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <T, R : Any> Iterable<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): List<R> {\n return mapIndexedNotNullTo(ArrayList<R>(), transform)\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original collection\n * and appends only the non-null results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <T, R : Any, C : MutableCollection<in R>> Iterable<T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {\n forEachIndexed { index, element -> transform(index, element)?.let { destination.add(it) } }\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original collection\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {\n var index = 0\n for (item in this)\n destination.add(transform(checkIndexOverflow(index++), item))\n return destination\n}\n\n/**\n * Returns a list containing only the non-null results of applying the given [transform] function\n * to each element in the original collection.\n * \n * @sample samples.collections.Collections.Transformations.mapNotNull\n */\npublic inline fun <T, R : Any> Iterable<T>.mapNotNull(transform: (T) -> R?): List<R> {\n return mapNotNullTo(ArrayList<R>(), transform)\n}\n\n/**\n * Applies the given [transform] function to each element in the original collection\n * and appends only the non-null results to the given [destination].\n */\npublic inline fun <T, R : Any, C : MutableCollection<in R>> Iterable<T>.mapNotNullTo(destination: C, transform: (T) -> R?): C {\n forEach { element -> transform(element)?.let { destination.add(it) } }\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original collection\n * and appends the results to the given [destination].\n */\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.mapTo(destination: C, transform: (T) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original collection\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\npublic fun <T> Iterable<T>.withIndex(): Iterable<IndexedValue<T>> {\n return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a list containing only distinct elements from the given collection.\n * \n * Among equal elements of the given collection, only the first one will be present in the resulting list.\n * The elements in the resulting list are in the same order as they were in the source collection.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun <T> Iterable<T>.distinct(): List<T> {\n return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only elements from the given collection\n * having distinct keys returned by the given [selector] function.\n * \n * Among elements of the given collection with equal keys, only the first one will be present in the resulting list.\n * The elements in the resulting list are in the same order as they were in the source collection.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic inline fun <T, K>

Iterable<T>.distinctBy(selector: (T) -> K): List<T> {\n val set = HashSet<K>()\n val list = ArrayList<T>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a set containing all elements that are contained by both this collection and the specified collection.\n * \n * The returned set preserves the element iteration order of the original collection.\n * \n * To get a set containing all elements that are contained at least in one of these collections use [union].\n */\npublic infix fun <T> Iterable<T>.intersect(other: Iterable<T>): Set<T> {\n val set = this.toMutableSet()\n set.retainAll(other)\n return set\n}\n\n/**\n * Returns a set containing all elements that are contained by this collection and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original collection.\n */\npublic infix fun <T> Iterable<T>.subtract(other: Iterable<T>): Set<T> {\n val set = this.toMutableSet()\n set.removeAll(other)\n return set\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given collection.\n * \n * The returned set preserves the element iteration order of the original collection.\n */\npublic fun <T> Iterable<T>.toMutableSet(): MutableSet<T> {\n return when (this) {\n is Collection<T> -> LinkedHashSet(this)\n else -> toCollection(LinkedHashSet<T>())\n }\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original collection.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n * \n * To get a set containing all elements that are contained in both collections use [intersect].\n */\npublic infix fun <T> Iterable<T>.union(other: Iterable<T>): Set<T> {\n val set = this.toMutableSet()\n set.addAll(other)\n return set\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun <T> Iterable<T>.all(predicate: (T) -> Boolean): Boolean {\n if (this is Collection && isEmpty()) return true\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if collection has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun <T> Iterable<T>.any(): Boolean {\n if (this is Collection) return !isEmpty()\n return iterator().hasNext()\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun <T> Iterable<T>.any(predicate: (T) -> Boolean): Boolean {\n if (this is Collection && isEmpty()) return false\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns the number of elements in this collection.\n */\npublic fun <T> Iterable<T>.count(): Int {\n if (this is Collection) return size\n var count = 0\n for (element in this) checkCountOverflow(++count)\n return count\n}\n\n/**\n * Returns the number of elements in this collection.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Collection<T>.count(): Int {\n return size\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun <T> Iterable<T>.count(predicate: (T) -> Boolean): Int {\n if (this is Collection && isEmpty()) return 0\n var count = 0\n for (element in this) if (predicate(element)) checkCountOverflow(++count)\n return count\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the collection is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <T, R> Iterable<T>.fold(initial: R, operation: (acc: R, T) -> R): R {\n var accumulator = initial\n for (element in this) accumulator = operation(accumulator, element)\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original collection.\n * \n * Returns the specified [initial] value if the collection is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <T, R> Iterable<T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {\n var index = 0\n var accumulator = initial\n for (element in this) accumulator = operation(checkIndexOverflow(index++), accumulator, element)\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the list is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n

*/\npublic inline fun <T, R> List<T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {\n var accumulator = initial\n if (!isEmpty()) {\n val iterator = listIterator(size)\n while (iterator.hasPrevious()) {\n accumulator = operation(iterator.previous(), accumulator)\n }\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original list and current accumulator value.\n * \n * Returns the specified [initial] value if the list is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <T, R> List<T>.foldRightIndexed(initial: R, operation: (index: Int, T, acc: R) -> R): R {\n var accumulator = initial\n if (!isEmpty()) {\n val iterator = listIterator(size)\n while (iterator.hasPrevious()) {\n val index = iterator.previousIndex()\n accumulator = operation(index, iterator.previous(), accumulator)\n }\n }\n return accumulator\n}\n\n/**\n * Performs the given [action] on each element.\n */\n@kotlin.internal.HidesMembers\npublic inline fun <T> Iterable<T>.forEach(action: (T) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\npublic inline fun <T> Iterable<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {\n var index = 0\n for (item in this) action(checkIndexOverflow(index++), item)\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Iterable<Double>.max(): Double? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Iterable<Float>.max(): Float? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T : Comparable<T>> Iterable<T>.max(): T? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <T, R : Comparable<R>> Iterable<T>.maxBy(selector: (T) -> R): T? {\n return maxByOrNull(selector)\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, R : Comparable<R>> Iterable<T>.maxByOrNull(selector: (T) -> R): T? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var maxElem = iterator.next()\n if (!iterator.hasNext()) return maxElem\n var maxValue = selector(maxElem)\n do {\n val e = iterator.next()\n val v = selector(e)\n if (maxValue < v) {\n maxElem = e\n maxValue = v\n }\n } while (iterator.hasNext())\n return maxElem\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the collection.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the collection is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.maxOf(selector: (T) -> Double): Double {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var maxValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the collection.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the collection is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.maxOf(selector: (T) ->

Float): Float {\n   val iterator = iterator()\n   if (!iterator.hasNext()) throw NoSuchElementException()\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the collection.\n * \n * @throws NoSuchElementException if the collection is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Iterable<T>.maxOf(selector: (T) -> R): R {\n   val iterator = iterator()\n   if (!iterator.hasNext()) throw NoSuchElementException()\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the collection or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.maxOfOrNull(selector: (T) -> Double): Double? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the collection or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.maxOfOrNull(selector: (T) -> Float): Float? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the collection or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Iterable<T>.maxOfOrNull(selector: (T) -> R): R? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the collection.\n * \n * @throws NoSuchElementException if the collection is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Iterable<T>.maxOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {\n   val iterator = iterator()\n   if (!iterator.hasNext()) throw NoSuchElementException()\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Iterable<T>.maxOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest element or `null` if there are no

elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun Iterable<Double>.maxOrNull(): Double? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var max = iterator.next()\n while (iterator.hasNext()) {\n val e = iterator.next()\n max = maxOf(max, e)\n }\n return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun Iterable<Float>.maxOrNull(): Float? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var max = iterator.next()\n while (iterator.hasNext()) {\n val e = iterator.next()\n max = maxOf(max, e)\n }\n return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T : Comparable<T>> Iterable<T>.maxOrNull(): T? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var max = iterator.next()\n while (iterator.hasNext()) {\n val e = iterator.next()\n if (max < e) max = e\n }\n return max\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T> Iterable<T>.maxWith(comparator: Comparator<in T>): T? {\n return maxWithOrNull(comparator)\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Iterable<T>.maxWithOrNull(comparator: Comparator<in T>): T? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var max = iterator.next()\n while (iterator.hasNext()) {\n val e = iterator.next()\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Iterable<Double>.min(): Double? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Iterable<Float>.min(): Float? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T : Comparable<T>> Iterable<T>.min(): T? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <T, R : Comparable<R>> Iterable<T>.minBy(selector: (T) -> R): T? {\n return minByOrNull(selector)\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, R : Comparable<R>> Iterable<T>.minByOrNull(selector: (T) -> R): T? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var minElem = iterator.next()\n if (!iterator.hasNext()) return minElem\n var minValue = selector(minElem)\n do {\n val e = iterator.next()\n val v = selector(e)\n if (minValue > v) {\n minElem = e\n minValue = v\n }\n } while (iterator.hasNext())\n return minElem\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the collection.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the collection is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.minOf(selector: (T) -> Double): Double {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the collection.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the collection is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.minOf(selector: (T) -> Float): Float {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the collection.\n * \n * @throws NoSuchElementException if the collection is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Iterable<T>.minOf(selector: (T) -> R): R {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the collection or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Double): Double? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the collection or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Float): Float? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the collection or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Iterable<T>.minOfOrNull(selector: (T) -> R): R? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the collection.\n * \n * @throws NoSuchElementException if the collection is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Iterable<T>.minOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Iterable<T>.minOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest element or `null` if there are

```
no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun Iterable<Double>.minOrNull(): Double? {\n    val iterator = iterator()\n    if (!iterator.hasNext()) return null\n    var min = iterator.next()\n    while (iterator.hasNext()) {\n        val e = iterator.next()\n        min = minOf(min, e)\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic fun Iterable<Float>.minOrNull(): Float? {\n    val iterator = iterator()\n    if (!iterator.hasNext()) return null\n    var min = iterator.next()\n    while (iterator.hasNext()) {\n        val e = iterator.next()\n        min = minOf(min, e)\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T : Comparable<T>> Iterable<T>.minOrNull(): T? {\n    val iterator = iterator()\n    if (!iterator.hasNext()) return null\n    var min = iterator.next()\n    while (iterator.hasNext()) {\n        val e = iterator.next()\n        if (min > e) min = e\n    }\n    return min\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T> Iterable<T>.minWith(comparator: Comparator<in T>): T? {\n    return minWithOrNull(comparator)\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Iterable<T>.minWithOrNull(comparator: Comparator<in T>): T? {\n    val iterator = iterator()\n    if (!iterator.hasNext()) return null\n    var min = iterator.next()\n    while (iterator.hasNext()) {\n        val e = iterator.next()\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n}\n\n/**\n * Returns `true` if the collection has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun <T> Iterable<T>.none(): Boolean {\n    if (this is Collection) return isEmpty()\n    return !iterator().hasNext()\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun <T> Iterable<T>.none(predicate: (T) -> Boolean): Boolean {\n    if (this is Collection && isEmpty()) return true\n    for (element in this) if (predicate(element)) return false\n    return true\n}\n\n/**\n * Performs the given [action] on each element and returns the collection itself afterwards.\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, C : Iterable<T>> C.onEach(action: (T) -> Unit): C {\n    return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the collection itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, C : Iterable<T>> C.onEachIndexed(action: (index: Int, T) -> Unit): C {\n    return apply { forEachIndexed(action) }\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this collection is empty. If the collection can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun <S, T : S> Iterable<T>.reduce(operation: (acc: S, T) -> S): S {\n    val iterator = this.iterator()\n    if (!iterator.hasNext()) throw UnsupportedOperationException(\"Empty collection can't be reduced.\")\n    var accumulator: S = iterator.next()\n    while (iterator.hasNext()) {\n        accumulator = operation(accumulator, iterator.next())\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original collection.\n * \n * Throws an exception if this collection is empty. If the collection can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun <S, T : S> Iterable<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {\n    val iterator = this.iterator()\n    if (!iterator.hasNext()) throw UnsupportedOperationException(\"Empty collection can't be reduced.\")\n    var index = 1\n    var accumulator: S = iterator.next()\n    while (iterator.hasNext()) {\n        accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())\n    }\n    return
```

```
accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original collection.\n * \n * Returns
`null` if the collection is empty.\n * \n * @param [operation] function that takes the index of an element, current
accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample
samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <S, T : S>
Iterable<T>.reduceIndexedOrNull(operation: (index: Int, acc: S, T) -> S): S? {\n    val iterator = this.iterator()\n    if
(!iterator.hasNext()) return null\n    var index = 1\n    var accumulator: S = iterator.next()\n    while
(iterator.hasNext()) {\n        accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())\n
}\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation]
from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the collection is
empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates
the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <S, T : S>
Iterable<T>.reduceOrNull(operation: (acc: S, T) -> S): S? {\n    val iterator = this.iterator()\n    if
(!iterator.hasNext()) return null\n    var accumulator: S = iterator.next()\n    while (iterator.hasNext()) {\n
accumulator = operation(accumulator, iterator.next())\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value
starting with the last element and applying [operation] from right to left\n * to each element and current accumulator
value.\n * \n * Throws an exception if this list is empty. If the list can be empty in an expected way,\n * please use
[reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that
takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample
samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun <S, T : S>
List<T>.reduceRight(operation: (T, acc: S) -> S): S {\n    val iterator = listIterator(size)\n    if
(!iterator.hasPrevious())\n        throw UnsupportedOperationException(\"Empty list can't be reduced.\")\n    var
accumulator: S = iterator.previous()\n    while (iterator.hasPrevious()) {\n        accumulator =
operation(iterator.previous(), accumulator)\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting
with the last element and applying [operation] from right to left\n * to each element with its index in the original list
and current accumulator value.\n * \n * Throws an exception if this list is empty. If the list can be empty in an
expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n
* @param [operation] function that takes the index of an element, the element itself and current accumulator
value,\n * and calculates the next accumulator value.\n * \n * @sample
samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun <S, T : S>
List<T>.reduceRightIndexed(operation: (index: Int, T, acc: S) -> S): S {\n    val iterator = listIterator(size)\n    if
(!iterator.hasPrevious())\n        throw UnsupportedOperationException(\"Empty list can't be reduced.\")\n    var
accumulator: S = iterator.previous()\n    while (iterator.hasPrevious()) {\n        val index = iterator.previousIndex()\n
    accumulator = operation(index, iterator.previous(), accumulator)\n    }\n    return accumulator\n}\n\n/**\n *
Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with
its index in the original list and current accumulator value.\n * \n * Returns `null` if the list is empty.\n * \n *
@param [operation] function that takes the index of an element, the element itself and current accumulator value,\n
* and calculates the next accumulator value.\n * \n * @sample
samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <S,
T : S> List<T>.reduceRightIndexedOrNull(operation: (index: Int, T, acc: S) -> S): S? {\n    val iterator =
listIterator(size)\n    if (!iterator.hasPrevious())\n        return null\n    var accumulator: S = iterator.previous()\n
while (iterator.hasPrevious()) {\n        val index = iterator.previousIndex()\n        accumulator = operation(index,
iterator.previous(), accumulator)\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the last
element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n *
Returns `null` if the list is empty.\n * \n * @param [operation] function that takes an element and current
accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample
samples.collections.Collections.Aggregates.reduceRightOrNull\n
```

```
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <S, T : S> List<T>.reduceRightOrNull(operation: (T, acc: S) -> S): S? {\n    val iterator = listIterator(size)\n    if (!iterator.hasPrevious())\n        return null\n    var accumulator: S = iterator.previous()\n    while (iterator.hasPrevious()) {\n        accumulator = operation(iterator.previous(), accumulator)\n    }\n    return accumulator\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, R> Iterable<T>.runningFold(initial: R, operation: (acc: R, T) -> R): List<R> {\n    val estimatedSize = collectionSizeOrDefault(9)\n    if (estimatedSize == 0) return listOf(initial)\n    val result = ArrayList<R>(estimatedSize + 1).apply { add(initial) }\n    var accumulator = initial\n    for (element in this) {\n        accumulator = operation(accumulator, element)\n        result.add(accumulator)\n    }\n    return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original collection and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, R> Iterable<T>.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {\n    val estimatedSize = collectionSizeOrDefault(9)\n    if (estimatedSize == 0) return listOf(initial)\n    val result = ArrayList<R>(estimatedSize + 1).apply { add(initial) }\n    var index = 0\n    var accumulator = initial\n    for (element in this) {\n        accumulator = operation(index++, accumulator, element)\n        result.add(accumulator)\n    }\n    return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this collection.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and the element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <S, T : S> Iterable<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {\n    val iterator = this.iterator()\n    if (!iterator.hasNext()) return emptyList()\n    var accumulator: S = iterator.next()\n    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply { add(accumulator) }\n    while (iterator.hasNext()) {\n        accumulator = operation(accumulator, iterator.next())\n        result.add(accumulator)\n    }\n    return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original collection and current accumulator value that starts with the first element of this collection.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <S, T : S> Iterable<T>.runningReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {\n    val iterator = this.iterator()\n    if (!iterator.hasNext()) return emptyList()\n    var accumulator: S = iterator.next()\n    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply { add(accumulator) }\n    var index = 1\n    while (iterator.hasNext()) {\n        accumulator = operation(index++, accumulator, iterator.next())\n        result.add(accumulator)\n    }\n    return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation]
```

function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <T, R> Iterable<T>.scan(initial: R, operation: (acc: R, T) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original collection and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <T, R> Iterable<T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {\n   return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the collection.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun <T> Iterable<T>.sumBy(selector: (T) -> Int): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the collection.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun <T> Iterable<T>.sumByDouble(selector: (T) -> Double): Double {\n   var sum: Double = 0.0\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the collection.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.sumOf(selector: (T) -> Double): Double {\n   var sum: Double = 0.toDouble()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the collection.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.sumOf(selector: (T) -> Int): Int {\n   var sum: Int = 0.toInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the collection.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.sumOf(selector: (T) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the collection.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.sumOf(selector: (T) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the collection.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.sumOf(selector: (T) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return

sum\n}\n\n/**\n * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.\n */\npublic fun <T : Any> Iterable<T?>.requireNoNulls(): Iterable<T> {\n for (element in this) {\n if (element == null) {\n throw IllegalArgumentException(\"null element found in $this.\")\n }\n }\n @Suppress(\"UNCHECKED_CAST\")\n return this as Iterable<T>\n}\n\n/**\n * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.\n */\npublic fun <T : Any> List<T?>.requireNoNulls(): List<T> {\n for (element in this) {\n if (element == null) {\n throw IllegalArgumentException(\"null element found in $this.\")\n }\n }\n @Suppress(\"UNCHECKED_CAST\")\n return this as List<T>\n}\n\n/**\n * Splits this collection into a list of lists each not exceeding the given [size].\n * \n * The last list in the resulting list may have fewer elements than the given [size].\n * \n * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.\n * \n * @sample samples.collections.Collections.Transformations.chunked\n */\n@SinceKotlin(\"1.2\")\npublic fun <T> Iterable<T>.chunked(size: Int): List<List<T>> {\n return windowed(size, size, partialWindows = true)\n}\n\n/**\n * Splits this collection into several lists each not exceeding the given [size]\n * and applies the given [transform] function to an each.\n * \n * @return list of results of the [transform] applied to an each list.\n * \n * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * The last list may have fewer elements than the given [size].\n * \n * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.\n * \n * @sample samples.text.Strings.chunkedTransform\n */\n@SinceKotlin(\"1.2\")\npublic fun <T, R> Iterable<T>.chunked(size: Int, transform: (List<T>) -> R): List<R> {\n return windowed(size, size, partialWindows = true, transform = transform)\n}\n\n/**\n * Returns a list containing all elements of the original collection without the first occurrence of the given [element].\n */\npublic operator fun <T> Iterable<T>.minus(element: T): List<T> {\n val result = ArrayList<T>(collectionSizeOrDefault(10))\n var removed = false\n return this.filterTo(result) { if (!removed && it == element) { removed = true; false } else true }\n}\n\n/**\n * Returns a list containing all elements of the original collection except the elements contained in the given [elements] array.\n * \n * Before Kotlin 1.6, the [elements] array may have been converted to a [HashSet] to speed up the operation, thus the elements were required to have\n * a correct and stable implementation of `hashCode()` that didn't change between successive invocations.\n * On JVM, you can enable this behavior back with the system property `kotlin.collections.convert_arg_to_set_in_removeAll` set to `true`.\n */\npublic operator fun <T> Iterable<T>.minus(elements: Array<out T>): List<T> {\n if (elements.isEmpty()) return this.toList()\n val other = elements.convertToSetForSetOperation()\n return this.filterNot { it in other }\n}\n\n/**\n * Returns a list containing all elements of the original collection except the elements contained in the given [elements] collection.\n * \n * Before Kotlin 1.6, the [elements] collection may have been converted to a [HashSet] to speed up the operation, thus the elements were required to have\n * a correct and stable implementation of `hashCode()` that didn't change between successive invocations.\n * On JVM, you can enable this behavior back with the system property `kotlin.collections.convert_arg_to_set_in_removeAll` set to `true`.\n */\npublic operator fun <T> Iterable<T>.minus(elements: Iterable<T>): List<T> {\n val other = elements.convertToSetForSetOperationWith(this)\n if (other.isEmpty())\n return this.toList()\n return this.filterNot { it in other }\n}\n\n/**\n * Returns a list containing all elements of the original collection except the elements contained in the given [elements] sequence.\n * \n * Before Kotlin 1.6, the [elements] sequence may have been converted to a [HashSet] to speed up the operation, thus the elements were required to have\n * a correct and stable implementation of `hashCode()` that didn't change between successive invocations.\n * On JVM, you can enable this behavior back with the system property `kotlin.collections.convert_arg_to_set_in_removeAll` set to `true`.\n */\npublic operator fun <T> Iterable<T>.minus(elements: Sequence<T>): List<T> {\n val other = elements.convertToSetForSetOperation()\n if (other.isEmpty())\n return this.toList()\n return this.filterNot { it in other }\n}\n\n/**\n * Returns a list containing all elements of the original collection without the first occurrence

of the given [element].\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.minusElement(element: T): List<T> {\n   return minus(element)\n}\n\n/**\n * Splits the original collection into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n * \n * @sample samples.collections.Iterables.Operations.partition\n */\npublic inline fun <T> Iterable<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {\n   val first = ArrayList<T>()\n   val second = ArrayList<T>()\n   for (element in this) {\n   if (predicate(element)) {\n   first.add(element)\n   } else {\n   second.add(element)\n   }\n   }\n   return Pair(first, second)\n}\n\n/**\n * Returns a list containing all elements of the original collection and then the given [element].\n */\npublic operator fun <T> Iterable<T>.plus(element: T): List<T> {\n   if (this is Collection) return this.plus(element)\n   val result = ArrayList<T>()\n   result.addAll(this)\n   result.add(element)\n   return result\n}\n\n/**\n * Returns a list containing all elements of the original collection and then the given [element].\n */\npublic operator fun <T> Collection<T>.plus(element: T): List<T> {\n   val result = ArrayList<T>(size + 1)\n   result.addAll(this)\n   result.add(element)\n   return result\n}\n\n/**\n * Returns a list containing all elements of the original collection and then all elements of the given [elements] array.\n */\npublic operator fun <T> Iterable<T>.plus(elements: Array<out T>): List<T> {\n   if (this is Collection) return this.plus(elements)\n   val result = ArrayList<T>()\n   result.addAll(this)\n   result.addAll(elements)\n   return result\n}\n\n/**\n * Returns a list containing all elements of the original collection and then all elements of the given [elements] array.\n */\npublic operator fun <T> Collection<T>.plus(elements: Array<out T>): List<T> {\n   val result = ArrayList<T>(this.size + elements.size)\n   result.addAll(this)\n   result.addAll(elements)\n   return result\n}\n\n/**\n * Returns a list containing all elements of the original collection and then all elements of the given [elements] collection.\n */\npublic operator fun <T> Iterable<T>.plus(elements: Iterable<T>): List<T> {\n   if (this is Collection) return this.plus(elements)\n   val result = ArrayList<T>()\n   result.addAll(this)\n   result.addAll(elements)\n   return result\n}\n\n/**\n * Returns a list containing all elements of the original collection and then all elements of the given [elements] collection.\n */\npublic operator fun <T> Collection<T>.plus(elements: Iterable<T>): List<T> {\n   if (elements is Collection) {\n   val result = ArrayList<T>(this.size + elements.size)\n   result.addAll(this)\n   result.addAll(elements)\n   return result\n   } else {\n   val result = ArrayList<T>(this)\n   result.addAll(elements)\n   return result\n   }\n}\n\n/**\n * Returns a list containing all elements of the original collection and then all elements of the given [elements] sequence.\n */\npublic operator fun <T> Iterable<T>.plus(elements: Sequence<T>): List<T> {\n   val result = ArrayList<T>()\n   result.addAll(this)\n   result.addAll(elements)\n   return result\n}\n\n/**\n * Returns a list containing all elements of the original collection and then all elements of the given [elements] sequence.\n */\npublic operator fun <T> Collection<T>.plus(elements: Sequence<T>): List<T> {\n   val result = ArrayList<T>(this.size + 10)\n   result.addAll(this)\n   result.addAll(elements)\n   return result\n}\n\n/**\n * Returns a list containing all elements of the original collection and then the given [element].\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.plusElement(element: T): List<T> {\n   return plus(element)\n}\n\n/**\n * Returns a list containing all elements of the original collection and then the given [element].\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Collection<T>.plusElement(element: T): List<T> {\n   return plus(element)\n}\n\n/**\n * Returns a list of snapshots of the window of the given [size]\n * sliding along this collection with the given [step], where each\n * snapshot is a list.\n * \n * Several last lists may have fewer elements than the given [size].\n * \n * Both [size] and [step] must be positive and can be greater than the number of elements in this collection.\n * @param size the number of elements to take in each window\n * @param step the number of elements to move the window forward by on an each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample samples.collections.Sequences.Transformations.takeWindows\n */\n@SinceKotlin(\"1.2\")\npublic fun <T> Iterable<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): List<List<T>> {\n   checkWindowSizeStep(size, step)\n   if (this is RandomAccess && this is List) {\n   val thisSize = this.size\n   val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1\n   val result =

```
ArrayList<List<T>>(resultCapacity)\n        var index = 0\n        while (index in 0 until thisSize) {\n            val windowSize = size.coerceAtMost(thisSize - index)\n            if (windowSize < size && !partialWindows) break\n            result.add(List(windowSize) { this[it + index] })\n            index += step\n        }\n        return result\n    }\n    val result = ArrayList<List<T>>()\n    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = false).forEach {\n        result.add(it)\n    }\n    return result\n}\n\n/**\n * Returns a list of results of applying the given [transform] function to\n * an each list representing a view over the window of the given [size]\n * sliding along this collection with the given [step].\n * \n * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * Several last lists may have fewer elements than the given [size].\n * \n * Both [size] and [step] must be positive and can be greater than the number of elements in this collection.\n * @param size the number of elements to take in each window\n * @param step the number of elements to move the window forward by on an each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample samples.collections.Sequences.Transformations.averageWindows\n */\n@SinceKotlin(\"1.2\")\npublic fun <T, R> Iterable<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (List<T>) -> R): List<R> {\n    checkWindowSizeStep(size, step)\n    if (this is RandomAccess && this is List) {\n        val thisSize = this.size\n        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1\n        val result = ArrayList<R>(resultCapacity)\n        val window = MovingSubList(this)\n        var index = 0\n        while (index in 0 until thisSize) {\n            val windowSize = size.coerceAtMost(thisSize - index)\n            if (!partialWindows && windowSize < size) break\n            window.move(index, index + windowSize)\n            result.add(transform(window))\n            index += step\n        }\n        return result\n    }\n    val result = ArrayList<R>()\n    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = true).forEach {\n        result.add(transform(it))\n    }\n    return result\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <T, R> Iterable<T>.zip(other: Array<out R>): List<Pair<T, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` collection and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <T, R, V> Iterable<T>.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {\n    val arraySize = other.size\n    val list = ArrayList<V>(minOf(collectionSizeOrDefault(10), arraySize))\n    var i = 0\n    for (element in this) {\n        if (i >= arraySize) break\n        list.add(transform(element, other[i++]))\n    }\n    return list\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] collection with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <T, R> Iterable<T>.zip(other: Iterable<R>): List<Pair<T, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` collection and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <T, R, V> Iterable<T>.zip(other: Iterable<R>, transform: (a: T, b: R) -> V): List<V> {\n    val first = iterator()\n    val second = other.iterator()\n    val list = ArrayList<V>(minOf(collectionSizeOrDefault(10), other.collectionSizeOrDefault(10)))\n    while (first.hasNext() && second.hasNext()) {\n        list.add(transform(first.next(), second.next()))\n    }\n    return list\n}\n\n/**\n * Returns a list of pairs of each two adjacent elements in this collection.\n * \n * The returned list is empty if this collection contains less than two elements.\n * \n * @sample samples.collections.Collections.Transformations.zipWithNext\n */\n@SinceKotlin(\"1.2\")\npublic fun <T> Iterable<T>.zipWithNext(): List<Pair<T, T>> {\n    return zipWithNext { a, b -> a to b }\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to an each pair of two adjacent elements in this collection.\n * \n * The returned list is empty if this collection contains
```

less than two elements.\n * \n * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas\n */\n@SinceKotlin(\"1.2\")\npublic inline fun <T, R> Iterable<T>.zipWithNext(transform: (a: T, b: T) -> R): List<R> {\n val iterator = iterator()\n if (!iterator.hasNext()) return emptyList()\n val result = mutableListOf<R>()\n var current = iterator.next()\n while (iterator.hasNext()) {\n val next = iterator.next()\n result.add(transform(current, next))\n current = next\n }\n return result\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <T, A : Appendable> Iterable<T>.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((T) -> CharSequence)? = null): A {\n buffer.append(prefix)\n var count = 0\n for (element in this) {\n if (++count > 1) buffer.append(separator)\n if (limit < 0 || count <= limit) {\n buffer.appendElement(element, transform)\n } else break\n }\n if (limit >= 0 && count > limit) buffer.append(truncated)\n buffer.append(postfix)\n return buffer\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n * \n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun <T> Iterable<T>.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((T) -> CharSequence)? = null): String {\n return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Returns this collection as an [Iterable].\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.asIterable(): Iterable<T> {\n return this\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original collection returning its elements when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromCollection\n */\npublic fun <T> Iterable<T>.asSequence(): Sequence<T> {\n return Sequence { this.iterator() }\n}\n\n/**\n * Returns an average value of elements in the collection.\n */\n@kotlin.jvm.JvmName(\"averageOfByte\")\npublic fun Iterable<Byte>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the collection.\n */\n@kotlin.jvm.JvmName(\"averageOfShort\")\npublic fun Iterable<Short>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the collection.\n */\n@kotlin.jvm.JvmName(\"averageOfInt\")\npublic fun Iterable<Int>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the collection.\n */\n@kotlin.jvm.JvmName(\"averageOfLong\")\npublic fun Iterable<Long>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the collection.\n */\n@kotlin.jvm.JvmName(\"averageOfFloat\")\npublic fun Iterable<Float>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the collection.\n */\n@kotlin.jvm.JvmName(\"averageOfDouble\")\npublic fun Iterable<Double>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns the sum of all elements in the collection.\n */\n@kotlin.jvm.JvmName(\"sumOfByte\")\npublic fun

Iterable<Byte>.sum(): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n */\n@kotlin.jvm.JvmName(\"sumOfShort\")\npublic fun Iterable<Short>.sum(): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n */\n@kotlin.jvm.JvmName(\"sumOfInt\")\npublic fun Iterable<Int>.sum(): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n */\n@kotlin.jvm.JvmName(\"sumOfLong\")\npublic fun Iterable<Long>.sum(): Long {\n   var sum: Long = 0L\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n */\n@kotlin.jvm.JvmName(\"sumOfFloat\")\npublic fun Iterable<Float>.sum(): Float {\n   var sum: Float = 0.0f\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n */\n@kotlin.jvm.JvmName(\"sumOfDouble\")\npublic fun Iterable<Double>.sum(): Double {\n   var sum: Double = 0.0\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\nimport kotlin.comparisons.naturalOrder\nimport kotlin.random.Random\n\n/**\n * Returns the array if it's not `null`, or an empty array otherwise.\n * @sample samples.collections.Arrays.Usage.arrayOrEmpty\n */\n@kotlin.internal.InlineOnly\npublic actual inline fun <T> Array<out T>?.orEmpty(): Array<out T> = this ?: emptyArray<T>()\n\n/**\n * Returns a *typed* array containing all of the elements of this collection.\n *\n * Allocates an array of runtime type `T` having its size equal to the size of this collection\n * and populates the array with the elements of this collection.\n * @sample samples.collections.Collections.Collections.collectionToTypedArray\n */\n@kotlin.internal.InlineOnly\npublic actual inline fun <T> Collection<T>.toTypedArray(): Array<T> = copyToArray(this)\n\n@JsName(\"copyToArray\")\n@PublishedApi\ninternal fun <T> copyToArray(collection: Collection<T>): Array<T> {\n   return if (collection.asDynamic().toArray !== undefined)\n   collection.asDynamic().toArray().unsafeCast<Array<T>>()\n   else\n   copyToArrayImpl(collection).unsafeCast<Array<T>>()\n}\n\n@JsName(\"copyToArrayImpl\")\ninternal actual fun copyToArrayImpl(collection: Collection<*>): Array<Any?> {\n   val array = emptyArray<Any?>()\n   val iterator = collection.iterator()\n   while (iterator.hasNext())\n   array.asDynamic().push(iterator.next())\n   return array\n}\n\n@JsName(\"copyToExistingArrayImpl\")\ninternal actual fun <T> copyToArrayImpl(collection: Collection<*>, array: Array<T>): Array<T> {\n   if (array.size < collection.size)\n   return copyToArrayImpl(collection).unsafeCast<Array<T>>()\n\n   val iterator = collection.iterator()\n   var index = 0\n   while (iterator.hasNext()) {\n   array[index++] = iterator.next().unsafeCast<T>()\n   }\n   if (index < array.size) {\n   array[index] = null.unsafeCast<T>()\n   }\n   return array\n}\n\n\n/**\n * Returns an immutable list containing only the specified object [element].\n */\npublic fun <T> listOf(element: T): List<T> = arrayListOf(element)\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal actual inline fun <E> buildListInternal(builderAction: MutableList<E>.() -> Unit): List<E> {\n   return ArrayList<E>().apply(builderAction).build()\n}\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal actual inline fun <E> buildListInternal(capacity: Int, builderAction: MutableList<E>.() -> Unit): List<E> {\n   checkBuilderCapacity(capacity)\n   return ArrayList<E>(capacity).apply(builderAction).build()\n}\n\n\n/**\n * Returns an immutable set containing only the specified object [element].\n */\npublic fun <T> setOf(element: T): Set<T> = hashSetOf(element)\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal actual inline fun <E> buildSetInternal(builderAction: MutableSet<E>.() -> Unit): Set<E> {\n   return LinkedHashSet<E>().apply(builderAction).build()\n}\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal actual inline fun <E> buildSetInternal(capacity: Int, builderAction: MutableSet<E>.() -> Unit): Set<E> {\n   return LinkedHashSet<E>(capacity).apply(builderAction).build()\n}\n\n\n/**\n * Returns an immutable map, mapping only the specified key to the\n * specified value.\n */\npublic fun <K, V> mapOf(pair:

```
Pair<K, V>): Map<K, V> =
hashMapOf(pair)\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninternal actual inline
fun <K, V> buildMapInternal(builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {\n    return
LinkedHashMap<K,
V>().apply(builderAction).build()\n}\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\ninte
rnal actual inline fun <K, V> buildMapInternal(capacity: Int, builderAction: MutableMap<K, V>.() -> Unit):
Map<K, V> {\n    return LinkedHashMap<K, V>(capacity).apply(builderAction).build()\n}\n\n\n/**\n * Fills the
list with the provided [value].\n *\n * Each element in the list gets replaced with the [value].\n
*/\n@SinceKotlin(\"1.2\")\npublic actual fun <T> MutableList<T>.fill(value: T): Unit {\n    for (index in
0..lastIndex) {\n        this[index] = value\n    }\n}\n\n/**\n * Randomly shuffles elements in this list.\n *\n * See:
https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n
*/\n@SinceKotlin(\"1.2\")\npublic actual fun <T> MutableList<T>.shuffle(): Unit = shuffle(Random)\n\n/**\n *
Returns a new list with the elements of this list randomly shuffled.\n */\n@SinceKotlin(\"1.2\")\npublic actual fun
<T> Iterable<T>.shuffled(): List<T> = toMutableList().apply { shuffle() }\n\n/**\n * Sorts elements in the list in-
place according to their natural sort order.\n *\n * The sort is _stable_. It means that equal elements preserve their
order relative to each other after sorting.\n *\n * @sample samples.collections.Collections.Sorting.sortMutableList\n
*/\npublic actual fun <T : Comparable<T>> MutableList<T>.sort(): Unit {\n    collectionsSort(this,
naturalOrder())\n}\n\n/**\n * Sorts elements in the list in-place according to the order specified with [comparator].\n
*\n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n *\n
* @sample samples.collections.Collections.Sorting.sortMutableListWith\n */\npublic actual fun <T>
MutableList<T>.sortWith(comparator: Comparator<in T>): Unit {\n    collectionsSort(this,
comparator)\n}\n\nprivate fun <T> collectionsSort(list: MutableList<T>, comparator: Comparator<in T>) {\n    if
(list.size <= 1) return\n\n    val array = copyToArray(list)\n    sortArrayWith(array, comparator)\n\n    for (i in 0 until
array.size) {\n        list[i] = array[i]\n    }\n}\n\ninternal actual fun <T> arrayOfNulls(reference: Array<T>, size: Int):
Array<T> {\n    return
arrayOfNulls<Any>(size).unsafeCast<Array<T>>()\n}\n\n@SinceKotlin(\"1.3\")\n@PublishedApi\n@JsName(\"arr
ayCopy\")\ninternal fun <T> arrayCopy(source: Array<out T>, destination: Array<in T>, destinationOffset: Int,
startIndex: Int, endIndex: Int) {\n    AbstractList.checkRangeIndexes(startIndex, endIndex, source.size)\n    val
rangeSize = endIndex - startIndex\n    AbstractList.checkRangeIndexes(destinationOffset, destinationOffset +
rangeSize, destination.size)\n\n    if (js(\"ArrayBuffer\").isView(destination) &&
js(\"ArrayBuffer\").isView(source)) {\n        val subrange = source.asDynamic().subarray(startIndex, endIndex)\n
 destination.asDynamic().set(subrange, destinationOffset)\n    } else {\n        if (source !== destination ||
destinationOffset <= startIndex) {\n            for (index in 0 until rangeSize) {\n
destination[destinationOffset + index] = source[startIndex + index]\n            }\n        } else {\n            for (index in
rangeSize - 1 downTo 0) {\n                destination[destinationOffset + index] = source[startIndex + index]\n
}\n        }\n    }\n}\n\n// no singleton map implementation in js, return map as
is\n@Suppress(\"NOTHING_TO_INLINE\")\ninternal actual inline fun <K, V> Map<K,
V>.toSingletonMapOrSelf(): Map<K, V> = this\n\n@Suppress(\"NOTHING_TO_INLINE\")\ninternal actual inline
fun <K, V> Map<out K, V>.toSingletonMap(): Map<K, V> =
this.toMutableMap()\n\n\n@Suppress(\"NOTHING_TO_INLINE\")\ninternal actual inline fun <T> Array<out
T>.copyToArrayOfAny(isVarargs: Boolean): Array<out Any?> =\n    if (isVarargs)\n    // no need to copy vararg
array in JS\n        this\n    else\n        this.copyOf()\n\n\n\n@PublishedApi\ninternal actual fun
checkIndexOverflow(index: Int): Int {\n    if (index < 0) {\n        throwIndexOverflow()\n    }\n    return
index\n}\n\n@PublishedApi\ninternal actual fun checkCountOverflow(count: Int): Int {\n    if (count < 0) {\n
throwCountOverflow()\n    }\n    return count\n}\n\n\n/**\n * JS map and set implementations do not make use of
capacities or load factors.\n */\n@PublishedApi\ninternal actual fun mapCapacity(expectedSize: Int) =
expectedSize\n\n/**\n * Checks a collection builder function capacity argument.\n * In JS no validation is made in
Map/Set constructor yet.\n */\n@SinceKotlin(\"1.3\")\n@PublishedApi\ninternal fun
```

checkBuilderCapacity(capacity: Int) {\n   require(capacity >= 0) { \"capacity must be non-negative.\" }\n}\n\ninternal actual fun brittleContainsOptimizationEnabled(): Boolean = false","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"CollectionsKt\")\n\npackage kotlin.collections\n\n\n/**\n * Returns the given iterator itself. This allows to use an instance of iterator in a `for` loop.\n * @sample samples.collections.Iterators.iterator\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> Iterator<T>.iterator(): Iterator<T> = this\n\n/**\n * Returns an [Iterator] that wraps each element produced by the original iterator\n * into an [IndexedValue] containing the index of that element and the element itself.\n *\n * @sample samples.collections.Iterators.withIndexIterator\n */\npublic fun <T> Iterator<T>.withIndex(): Iterator<IndexedValue<T>> = IndexingIterator(this)\n\n/**\n * Performs the given [operation] on each element of this [Iterator].\n * @sample samples.collections.Iterators.forEachIterator\n */\npublic inline fun <T> Iterator<T>.forEach(operation: (T) -> Unit): Unit {\n   for (element in this) operation(element)\n}\n\n/**\n * Iterator transforming original `iterator` into iterator of [IndexedValue], counting index from zero.\n */\ninternal class IndexingIterator<out T>(private val iterator: Iterator<T>) : Iterator<IndexedValue<T>> {\n   private var index = 0\n   final override fun hasNext(): Boolean = iterator.hasNext()\n   final override fun next(): IndexedValue<T> = IndexedValue(checkIndexOverflow(index++), iterator.next())\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"ComparisonsKt\")\n\npackage kotlin.comparisons\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\n\n/**\n * Returns the greater of two values.\n * \n * If values are equal, returns the first one.\n */\n@SinceKotlin(\"1.1\")\npublic expect fun <T : Comparable<T>> maxOf(a: T, b: T): T\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Byte, b: Byte): Byte\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Short, b: Short): Short\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Int, b: Int): Int\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Long, b: Long): Long\n\n/**\n * Returns the greater of two values.\n * \n * If either value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Float, b: Float): Float\n\n/**\n * Returns the greater of two values.\n * \n * If either value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Double, b: Double): Double\n\n/**\n * Returns the greater of three values.\n * \n * If there are multiple equal maximal values, returns the first of them.\n */\n@SinceKotlin(\"1.1\")\npublic expect fun <T : Comparable<T>> maxOf(a: T, b: T, c: T): T\n\n/**\n * Returns the greater of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Byte, b: Byte, c: Byte): Byte\n\n/**\n * Returns the greater of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Short, b: Short, c: Short): Short\n\n/**\n * Returns the greater of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Int, b: Int, c: Int): Int\n\n/**\n * Returns the greater of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Long, b: Long, c: Long): Long\n\n/**\n * Returns the greater of three values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Float, b: Float, c: Float): Float\n\n/**\n * Returns the greater of three values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Double, b: Double, c: Double): Double\n\n/**\n * Returns the greater of three values according to the order specified by the given [comparator].\n * \n * If there are multiple equal maximal values, returns the first of them.\n */\n@SinceKotlin(\"1.1\")\npublic fun <T> maxOf(a: T, b: T, c: T, comparator: Comparator<in T>): T {\n   return

maxOf(a, maxOf(b, c, comparator), comparator)\n}\n\n/**\n * Returns the greater of two values according to the order specified by the given [comparator].\n * \n * If values are equal, returns the first one.\n */\n@SinceKotlin(\"1.1\")\npublic fun <T> maxOf(a: T, b: T, comparator: Comparator<in T>): T {\n   return if (comparator.compare(a, b) >= 0) a else b\n}\n\n/**\n * Returns the greater of the given values.\n * \n * If there are multiple equal maximal values, returns the first of them.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun <T : Comparable<T>> maxOf(a: T, vararg other: T): T\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun maxOf(a: Byte, vararg other: Byte): Byte\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun maxOf(a: Short, vararg other: Short): Short\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun maxOf(a: Int, vararg other: Int): Int\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun maxOf(a: Long, vararg other: Long): Long\n\n/**\n * Returns the greater of the given values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun maxOf(a: Float, vararg other: Float): Float\n\n/**\n * Returns the greater of the given values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun maxOf(a: Double, vararg other: Double): Double\n\n/**\n * Returns the greater of the given values according to the order specified by the given [comparator].\n * \n * If there are multiple equal maximal values, returns the first of them.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> maxOf(a: T, vararg other: T, comparator: Comparator<in T>): T {\n   var max = a\n   for (e in other) if (comparator.compare(max, e) < 0) max = e\n   return max\n}\n\n/**\n * Returns the smaller of two values.\n * \n * If values are equal, returns the first one.\n */\n@SinceKotlin(\"1.1\")\npublic expect fun <T : Comparable<T>> minOf(a: T, b: T): T\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Byte, b: Byte): Byte\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Short, b: Short): Short\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Int, b: Int): Int\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Long, b: Long): Long\n\n/**\n * Returns the smaller of two values.\n * \n * If either value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Float, b: Float): Float\n\n/**\n * Returns the smaller of two values.\n * \n * If either value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Double, b: Double): Double\n\n/**\n * Returns the smaller of three values.\n * \n * If there are multiple equal minimal values, returns the first of them.\n */\n@SinceKotlin(\"1.1\")\npublic expect fun <T : Comparable<T>> minOf(a: T, b: T, c: T): T\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Byte, b: Byte, c: Byte): Byte\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Short, b: Short, c: Short): Short\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Int, b: Int, c: Int): Int\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Long, b: Long, c: Long): Long\n\n/**\n * Returns the smaller of three values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Float, b: Float, c: Float): Float\n\n/**\n * Returns the smaller of three values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Double, b: Double, c: Double): Double\n\n/**\n * Returns the smaller of three values according to the order specified by the given [comparator].\n * \n * If there are multiple equal minimal values, returns the first of them.\n */\n@SinceKotlin(\"1.1\")\npublic fun <T> minOf(a: T, b: T, c: T, comparator: Comparator<in T>): T {\n   return minOf(a, minOf(b, c, comparator), comparator)\n}\n\n/**\n * Returns the smaller of two values according to the order specified by the given [comparator].\n * \n * If values are equal, returns the first one.\n */\n@SinceKotlin(\"1.1\")\npublic fun <T> minOf(a: T, b: T, comparator: Comparator<in T>): T {\n   return if

(comparator.compare(a, b) <= 0) a else b\n}\n\n/**\n * Returns the smaller of the given values.\n * \n * If there are multiple equal minimal values, returns the first of them.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun <T : Comparable<T>> minOf(a: T, vararg other: T): T\n\n/**\n * Returns the smaller of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun minOf(a: Byte, vararg other: Byte): Byte\n\n/**\n * Returns the smaller of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun minOf(a: Short, vararg other: Short): Short\n\n/**\n * Returns the smaller of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun minOf(a: Int, vararg other: Int): Int\n\n/**\n * Returns the smaller of the given values.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun minOf(a: Long, vararg other: Long): Long\n\n/**\n * Returns the smaller of the given values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun minOf(a: Float, vararg other: Float): Float\n\n/**\n * Returns the smaller of the given values.\n * \n * If any value is `NaN`, returns `NaN`.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun minOf(a: Double, vararg other: Double): Double\n\n/**\n * Returns the smaller of the given values according to the order specified by the given [comparator].\n * \n * If there are multiple equal minimal values, returns the first of them.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> minOf(a: T, vararg other: T, comparator: Comparator<in T>): T {\n var min = a\n for (e in other) if (comparator.compare(min, e) > 0) min = e\n return min\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"MapsKt\")\n\npackage kotlin.collections\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\nimport kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns the first non-null value produced by [transform] function being applied to entries of this map in iteration order,\n * or throws [NoSuchElementException] if no non-null value was produced.\n * \n * @sample samples.collections.Collections.Transformations.firstNotNullOf\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R : Any> Map<out K, V>.firstNotNullOf(transform: (Map.Entry<K, V>) -> R?): R {\n return firstNotNullOfOrNull(transform) ?: throw NoSuchElementException(\"No element of the map was transformed to a non-null value.\")\n}\n\n/**\n * Returns the first non-null value produced by [transform] function being applied to entries of this map in iteration order,\n * or `null` if no non-null value was produced.\n * \n * @sample samples.collections.Collections.Transformations.firstNotNullOf\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R : Any> Map<out K, V>.firstNotNullOfOrNull(transform: (Map.Entry<K, V>) -> R?): R? {\n for (element in this) {\n val result = transform(element)\n if (result != null) {\n return result\n }\n }\n return null\n}\n\n/**\n * Returns a [List] containing all key-value pairs.\n */\npublic fun <K, V> Map<out K, V>.toList(): List<Pair<K, V>> {\n if (size == 0)\n return emptyList()\n val iterator = entries.iterator()\n if (!iterator.hasNext())\n return emptyList()\n val first = iterator.next()\n if (!iterator.hasNext())\n return listOf(first.toPair())\n val result = ArrayList<Pair<K, V>>(size)\n result.add(first.toPair())\n do {\n result.add(iterator.next().toPair())\n } while (iterator.hasNext())\n return result\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each entry of original map.\n * \n * @sample samples.collections.Maps.Transformations.flatMap\n */\npublic inline fun <K, V, R> Map<out K, V>.flatMap(transform: (Map.Entry<K, V>) -> Iterable<R>): List<R> {\n return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each entry of original map.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapSequence\")\npublic inline fun <K, V, R> Map<out K, V>.flatMap(transform: (Map.Entry<K, V>) -> Sequence<R>): List<R> {\n return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each entry of original map, to the given [destination].\n */\npublic inline fun <K, V, R, C : MutableCollection<in R>>

Map<out K, V>.flatMapTo(destination: C, transform: (Map.Entry<K, V>) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each entry of original map, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapSequenceTo\")\npublic inline fun <K, V, R, C : MutableCollection<in R>> Map<out K, V>.flatMapTo(destination: C, transform: (Map.Entry<K, V>) -> Sequence<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each entry in the original map.\n * \n * @sample samples.collections.Maps.Transformations.mapToList\n */\npublic inline fun <K, V, R> Map<out K, V>.map(transform: (Map.Entry<K, V>) -> R): List<R> {\n return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing only the non-null results of applying the given [transform] function\n * to each entry in the original map.\n * \n * @sample samples.collections.Maps.Transformations.mapNotNull\n */\npublic inline fun <K, V, R : Any> Map<out K, V>.mapNotNull(transform: (Map.Entry<K, V>) -> R?): List<R> {\n return mapNotNullTo(ArrayList<R>(), transform)\n}\n\n/**\n * Applies the given [transform] function to each entry in the original map\n * and appends only the non-null results to the given [destination].\n */\npublic inline fun <K, V, R : Any, C : MutableCollection<in R>> Map<out K, V>.mapNotNullTo(destination: C, transform: (Map.Entry<K, V>) -> R?): C {\n forEach { element -> transform(element)?.let { destination.add(it) } }\n return destination\n}\n\n/**\n * Applies the given [transform] function to each entry of the original map\n * and appends the results to the given [destination].\n */\npublic inline fun <K, V, R, C : MutableCollection<in R>> Map<out K, V>.mapTo(destination: C, transform: (Map.Entry<K, V>) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Returns `true` if all entries match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun <K, V> Map<out K, V>.all(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {\n if (isEmpty()) return true\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if map has at least one entry.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun <K, V> Map<out K, V>.any(): Boolean {\n return !isEmpty()\n}\n\n/**\n * Returns `true` if at least one entry matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun <K, V> Map<out K, V>.any(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {\n if (isEmpty()) return false\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns the number of entries in this map.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.count(): Int {\n return size\n}\n\n/**\n * Returns the number of entries matching the given [predicate].\n */\npublic inline fun <K, V> Map<out K, V>.count(predicate: (Map.Entry<K, V>) -> Boolean): Int {\n if (isEmpty()) return 0\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Performs the given [action] on each entry.\n */\n@kotlin.internal.HidesMembers\npublic inline fun <K, V> Map<out K, V>.forEach(action: (Map.Entry<K, V>) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R : Comparable<R>> Map<out K, V>.maxBy(selector: (Map.Entry<K, V>) -> R): Map.Entry<K, V>? {\n return maxByOrNull(selector)\n}\n\n/**\n * Returns the first entry yielding the largest value of the given function or `null` if there are no entries.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R : Comparable<R>> Map<out K, V>.maxByOrNull(selector: (Map.Entry<K, V>) -> R): Map.Entry<K, V>? {\n return entries.maxByOrNull(selector)\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each entry in the map.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the map is empty.\n

*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.maxOf(selector: (Map.Entry<K, V>) -> Double): Double {\n   return entries.maxOf(selector)\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each entry in the map.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the map is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.maxOf(selector: (Map.Entry<K, V>) -> Float): Float {\n   return entries.maxOf(selector)\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each entry in the map.\n * \n * @throws NoSuchElementException if the map is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R : Comparable<R>> Map<out K, V>.maxOf(selector: (Map.Entry<K, V>) -> R): R {\n   return entries.maxOf(selector)\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each entry in the map or `null` if there are no entries.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.maxOfOrNull(selector: (Map.Entry<K, V>) -> Double): Double? {\n   return entries.maxOfOrNull(selector)\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each entry in the map or `null` if there are no entries.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.maxOfOrNull(selector: (Map.Entry<K, V>) -> Float): Float? {\n   return entries.maxOfOrNull(selector)\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each entry in the map or `null` if there are no entries.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R : Comparable<R>> Map<out K, V>.maxOfOrNull(selector: (Map.Entry<K, V>) -> R): R? {\n   return entries.maxOfOrNull(selector)\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each entry in the map.\n * \n * @throws NoSuchElementException if the map is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R> Map<out K, V>.maxOfWith(comparator: Comparator<in R>, selector: (Map.Entry<K, V>) -> R): R {\n   return entries.maxOfWith(comparator, selector)\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each entry in the map or `null` if there are no entries.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R> Map<out K, V>.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Map.Entry<K, V>) -> R): R? {\n   return entries.maxOfWithOrNull(comparator, selector)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.maxWith(comparator: Comparator<in Map.Entry<K, V>>): Map.Entry<K, V>? {\n   return maxWithOrNull(comparator)\n}\n\n/**\n * Returns the first entry having the largest value according to the provided [comparator] or `null` if there are no entries.\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.maxWithOrNull(comparator: Comparator<in Map.Entry<K, V>>):

Map.Entry<K, V>? {\n   return entries.maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <K, V, R : Comparable<R>> Map<out K, V>.minBy(selector: (Map.Entry<K, V>) -> R): Map.Entry<K, V>? {\n   return minByOrNull(selector)\n}\n\n/**\n * Returns the first entry yielding the smallest value of the given function or `null` if there are no entries.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R : Comparable<R>> Map<out K, V>.minByOrNull(selector: (Map.Entry<K, V>) -> R): Map.Entry<K, V>? {\n   return entries.minByOrNull(selector)\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each entry in the map.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the map is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.minOf(selector: (Map.Entry<K, V>) -> Double): Double {\n   return entries.minOf(selector)\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each entry in the map.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the map is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.minOf(selector: (Map.Entry<K, V>) -> Float): Float {\n   return entries.minOf(selector)\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each entry in the map.\n * \n * @throws NoSuchElementException if the map is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R : Comparable<R>> Map<out K, V>.minOf(selector: (Map.Entry<K, V>) -> R): R {\n   return entries.minOf(selector)\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each entry in the map or `null` if there are no entries.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.minOfOrNull(selector: (Map.Entry<K, V>) -> Double): Double? {\n   return entries.minOfOrNull(selector)\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each entry in the map or `null` if there are no entries.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.minOfOrNull(selector: (Map.Entry<K, V>) -> Float): Float? {\n   return entries.minOfOrNull(selector)\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each entry in the map or `null` if there are no entries.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R : Comparable<R>> Map<out K, V>.minOfOrNull(selector: (Map.Entry<K, V>) -> R): R? {\n   return entries.minOfOrNull(selector)\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each entry in the map.\n * \n * @throws NoSuchElementException if the map is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R> Map<out K, V>.minOfWith(comparator: Comparator<in R>, selector: (Map.Entry<K, V>) -> R): R {\n   return entries.minOfWith(comparator, selector)\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each entry in the map or `null` if there

are no entries.\n
*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, R> Map<out K,
V>.minOfWithOrNull(comparator: Comparator<in R>, selector: (Map.Entry<K, V>) -> R): R? {\n   return
entries.minOfWithOrNull(comparator, selector)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\",
ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince
= \"1.5\", hiddenSince = \"1.6\")\npublic fun <K, V> Map<out K, V>.minWith(comparator: Comparator<in
Map.Entry<K, V>>): Map.Entry<K, V>? {\n   return minWithOrNull(comparator)\n}\n\n/**\n * Returns the first
entry having the smallest value according to the provided [comparator] or `null` if there are no entries.\n
*/\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K,
V>.minWithOrNull(comparator: Comparator<in Map.Entry<K, V>>): Map.Entry<K, V>? {\n   return
entries.minWithOrNull(comparator)\n}\n\n/**\n * Returns `true` if the map has no entries.\n * \n * @sample
samples.collections.Collections.Aggregates.none\n */\npublic fun <K, V> Map<out K, V>.none(): Boolean {\n
return isEmpty()\n}\n\n/**\n * Returns `true` if no entries match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun <K, V> Map<out K,
V>.none(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {\n   if (isEmpty()) return true\n   for (element in
this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Performs the given [action] on each entry and
returns the map itself afterwards.\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <K, V, M : Map<out K, V>>
M.onEach(action: (Map.Entry<K, V>) -> Unit): M {\n   return apply { for (element in this) action(element)
}\n}\n\n/**\n * Performs the given [action] on each entry, providing sequential index with the entry,\n * and returns
the map itself afterwards.\n * @param [action] function that takes the index of an entry and the entry itself\n * and
performs the action on the entry.\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <K, V, M : Map<out K, V>>
M.onEachIndexed(action: (index: Int, Map.Entry<K, V>) -> Unit): M {\n   return apply {
entries.forEachIndexed(action) }\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original map returning
its entries when being iterated.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K,
V>.asIterable(): Iterable<Map.Entry<K, V>> {\n   return entries\n}\n\n/**\n * Creates a [Sequence] instance that
wraps the original map returning its entries when being iterated.\n */\npublic fun <K, V> Map<out K,
V>.asSequence(): Sequence<Map.Entry<K, V>> {\n   return entries.asSequence()\n}\n\n","/*\n * Copyright 2010-
2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n//\n// NOTE:
THIS FILE IS AUTO-GENERATED by the GenerateUnicodeData.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\n// 10 mappings totally\ninternal fun
Char.titlecaseImpl(): String {\n   val uppercase = uppercase()\n   if (uppercase.length > 1) {\n   return if (this ==
'\\u0149') uppercase else uppercase[0] + uppercase.substring(1).lowercase()\n   }\n   return
titlecaseChar().toString()\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language
contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n/**\n * Converts this character to lower case using Unicode
mapping rules of the invariant locale.\n */\n@Deprecated(\"Use lowercaseChar() instead.\",
ReplaceWith(\"lowercaseChar()\"))\n@DeprecatedSinceKotlin(warningSince =
\"1.5\")\n@kotlin.internal.InlineOnly\npublic actual inline fun Char.toLowerCase(): Char =
lowercaseChar()\n\n/**\n * Converts this character to lower case using Unicode mapping rules of the invariant
locale.\n *\n * This function performs one-to-one character mapping.\n * To support one-to-many character
mapping use the [lowercase] function.\n * If this character has no mapping equivalent, the character itself is
returned.\n *\n * @sample samples.text.Chars.lowercase\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli
c actual inline fun Char.lowercaseChar(): Char = lowercase()[0]\n\n/**\n * Converts this character to lower case
using Unicode mapping rules of the invariant locale.\n *\n * This function supports one-to-many character mapping,
thus the length of the returned string can be greater than one.\n * For example, `'\\u0130'.lowercase()` returns

`\"\\u0069\\u0307\"`,\n * where `\\u0130'` is the LATIN CAPITAL LETTER I WITH DOT ABOVE character (`\u0130`).\n * If this character has no lower case mapping, the result of `toString()` of this char is returned.\n *\n * @sample samples.text.Chars.lowercase\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic actual inline fun Char.lowercase(): String = toString().asDynamic().toLowerCase().unsafeCast<String>()\n\n/**\n * Converts this character to upper case using Unicode mapping rules of the invariant locale.\n */\n@Deprecated(\"Use uppercaseChar() instead.\", ReplaceWith(\"uppercaseChar()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\n@kotlin.internal.InlineOnly\npublic actual inline fun Char.toUpperCase(): Char = uppercaseChar()\n\n/**\n * Converts this character to upper case using Unicode mapping rules of the invariant locale.\n *\n * This function performs one-to-one character mapping.\n * To support one-to-many character mapping use the [uppercase] function.\n * If this character has no mapping equivalent, the character itself is returned.\n *\n * @sample samples.text.Chars.uppercase\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Char.uppercaseChar(): Char {\n   val uppercase = uppercase()\n   return if (uppercase.length > 1) this else uppercase[0]\n}\n\n/**\n * Converts this character to upper case using Unicode mapping rules of the invariant locale.\n *\n * This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.\n * For example, `\\uFB00'.uppercase()` returns `\"\\u0046\\u0046\"`,\n * where `\\uFB00'` is the LATIN SMALL LIGATURE FF character (`\ufb00`).\n * If this character has no upper case mapping, the result of `toString()` of this char is returned.\n *\n * @sample samples.text.Chars.uppercase\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic actual inline fun Char.uppercase(): String = toString().asDynamic().toUpperCase().unsafeCast<String>()\n\n/**\n * Converts this character to title case using Unicode mapping rules of the invariant locale.\n *\n * This function performs one-to-one character mapping.\n * To support one-to-many character mapping use the [titlecase] function.\n * If this character has no mapping equivalent, the result of calling [uppercaseChar] is returned.\n *\n * @sample samples.text.Chars.titlecase\n */\n@SinceKotlin(\"1.5\")\npublic actual fun Char.titlecaseChar(): Char = titlecaseCharImpl()\n\n/**\n * Returns `true` if this character is a Unicode high-surrogate code unit (also known as leading-surrogate code unit).\n */\npublic actual fun Char.isHighSurrogate(): Boolean = this in Char.MIN_HIGH_SURROGATE..Char.MAX_HIGH_SURROGATE\n\n/**\n * Returns `true` if this character is a Unicode low-surrogate code unit (also known as trailing-surrogate code unit).\n */\npublic actual fun Char.isLowSurrogate(): Boolean = this in Char.MIN_LOW_SURROGATE..Char.MAX_LOW_SURROGATE\n\n/**\n * Returns the Unicode general category of this character.\n */\n@SinceKotlin(\"1.5\")\npublic actual val Char.category: CharCategory\n   get() = CharCategory.valueOf(getCategoryValue())\n\n/**\n * Returns `true` if this character (Unicode code point) is defined in Unicode.\n *\n * A character is considered to be defined in Unicode if its [category] is not [CharCategory.UNASSIGNED].\n */\n@SinceKotlin(\"1.5\")\npublic actual fun Char.isDefined(): Boolean {\n   if (this < '\\u0080') {\n   return true\n   }\n   return getCategoryValue() != CharCategory.UNASSIGNED.value\n}\n\n/**\n * Returns `true` if this character is a letter.\n *\n * A character is considered to be a letter if its [category] is [CharCategory.UPPERCASE_LETTER],\n * [CharCategory.LOWERCASE_LETTER], [CharCategory.TITLECASE_LETTER], [CharCategory.MODIFIER_LETTER], or [CharCategory.OTHER_LETTER].\n *\n * @sample samples.text.Chars.isLetter\n */\n@SinceKotlin(\"1.5\")\npublic actual fun Char.isLetter(): Boolean {\n   if (this in 'a'..'z' || this in 'A'..'Z') {\n   return true\n   }\n   if (this < '\\u0080') {\n   return false\n   }\n   return isLetterImpl()\n}\n\n/**\n * Returns `true` if this character is a letter or digit.\n *\n * @see isLetter\n * @see isDigit\n *\n * @sample samples.text.Chars.isLetterOrDigit\n */\n@SinceKotlin(\"1.5\")\npublic actual fun Char.isLetterOrDigit(): Boolean {\n   if (this in 'a'..'z' || this in 'A'..'Z' || this in '0'..'9') {\n   return true\n   }\n   if (this < '\\u0080') {\n   return false\n   }\n\n   return isDigitImpl() || isLetterImpl()\n}\n\n/**\n * Returns `true` if this character is a digit.\n *\n * A character is considered to be a digit if its [category] is

[CharCategory.DECIMAL_DIGIT_NUMBER].\n *\n * @sample samples.text.Chars.isDigit\n */\n@SinceKotlin(\"1.5\")\npublic actual fun Char.isDigit(): Boolean {\n if (this in '0'..'9') {\n return true\n }\n if (this < '\\u0080') {\n return false\n }\n return isDigitImpl()\n}\n\n/**\n * Returns `true` if this character is upper case.\n *\n * A character is considered to be an upper case character if its [category] is [CharCategory.UPPERCASE_LETTER],\n * or it has contributory property `Other_Uppercase` as defined by the Unicode Standard.\n *\n * @sample samples.text.Chars.isUpperCase\n */\n@SinceKotlin(\"1.5\")\npublic actual fun Char.isUpperCase(): Boolean {\n if (this in 'A'..'Z') {\n return true\n }\n if (this < '\\u0080') {\n return false\n }\n return isUpperCaseImpl()\n}\n\n/**\n * Returns `true` if this character is lower case.\n *\n * A character is considered to be a lower case character if its [category] is [CharCategory.LOWERCASE_LETTER],\n * or it has contributory property `Other_Lowercase` as defined by the Unicode Standard.\n *\n * @sample samples.text.Chars.isLowerCase\n */\n@SinceKotlin(\"1.5\")\npublic actual fun Char.isLowerCase(): Boolean {\n if (this in 'a'..'z') {\n return true\n }\n if (this < '\\u0080') {\n return false\n }\n return isLowerCaseImpl()\n}\n\n/**\n * Returns `true` if this character is a title case letter.\n *\n * A character is considered to be a title case letter if its [category] is [CharCategory.TITLECASE_LETTER].\n *\n * @sample samples.text.Chars.isTitleCase\n */\n@SinceKotlin(\"1.5\")\npublic actual fun Char.isTitleCase(): Boolean {\n if (this < '\\u0080') {\n return false\n }\n return getCategoryValue() == CharCategory.TITLECASE_LETTER.value\n}\n\n/**\n * Returns `true` if this character is an ISO control character.\n *\n * A character is considered to be an ISO control character if its [category] is [CharCategory.CONTROL],\n * meaning the Char is in the range `'\\u0000'..'\\u001F'` or in the range `'\\u007F'..'\\u009F'`.\n *\n * @sample samples.text.Chars.isISOControl\n */\n@SinceKotlin(\"1.5\")\npublic actual fun Char.isISOControl(): Boolean {\n return this <= '\\u001F' || this in '\\u007F'..'\\u009F'\n}\n\n/**\n * Determines whether a character is whitespace according to the Unicode standard.\n * Returns `true` if the character is whitespace.\n *\n * @sample samples.text.Chars.isWhitespace\n */\npublic actual fun Char.isWhitespace(): Boolean = isWhitespaceImpl()","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\nimport kotlin.js.RegExp\n\n/**\n * Converts the characters in the specified array to a string.\n */\n@SinceKotlin(\"1.2\")\n@Deprecated(\"Use CharArray.concatToString() instead\", ReplaceWith(\"chars.concatToString()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\")\npublic actual fun String(chars: CharArray): String {\n var result = \"\"\n for (char in chars) {\n result += char\n }\n return result\n}\n\n/**\n * Converts the characters from a portion of the specified array to a string.\n *\n * @throws IndexOutOfBoundsException if either [offset] or [length] are less than zero\n * or `offset + length` is out of [chars] array bounds.\n */\n@SinceKotlin(\"1.2\")\n@Deprecated(\"Use CharArray.concatToString(startIndex, endIndex) instead\", ReplaceWith(\"chars.concatToString(offset, offset + length)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\")\npublic actual fun String(chars: CharArray, offset: Int, length: Int): String {\n if (offset < 0 || length < 0 || chars.size - offset < length)\n throw IndexOutOfBoundsException(\"size: ${chars.size}; offset: $offset; length: $length\")\n var result = \"\"\n for (index in offset until offset + length) {\n result += chars[index]\n }\n return result\n}\n\n/**\n * Concatenates characters in this [CharArray] into a String.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun CharArray.concatToString(): String {\n var result = \"\"\n for (char in this) {\n result += char\n }\n return result\n}\n\n/**\n * Concatenates characters in this [CharArray] or its subrange into a String.\n *\n * @param startIndex the beginning (inclusive) of the subrange of characters, 0 by default.\n * @param endIndex the end (exclusive) of the subrange of characters, size of this array by default.\n *\n * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun CharArray.concatToString(startIndex: Int = 0, endIndex: Int = this.size): String {\n AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)\n var

result = \"\"\n   for (index in startIndex until endIndex) {\n   result += this[index]\n   }\n   return result\n}\n\n/**\n * Returns a [CharArray] containing characters of this string.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun String.toCharArray(): CharArray {\n   return CharArray(length) { get(it) }\n}\n\n/**\n * Returns a [CharArray] containing characters of this string or its substring.\n *\n * @param startIndex the beginning (inclusive) of the substring, 0 by default.\n * @param endIndex the end (exclusive) of the substring, length of this string by default.\n *\n * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the length of this string.\n * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun String.toCharArray(startIndex: Int = 0, endIndex: Int = this.length): CharArray {\n   AbstractList.checkBoundsIndexes(startIndex, endIndex, length)\n   return CharArray(endIndex - startIndex) { get(startIndex + it) }\n}\n\n/**\n * Decodes a string from the bytes in UTF-8 encoding in this array.\n *\n * Malformed byte sequences are replaced by the replacement char `\\uFFFD`.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun ByteArray.decodeToString(): String {\n   return decodeUtf8(this, 0, size, false)\n}\n\n/**\n * Decodes a string from the bytes in UTF-8 encoding in this array or its subrange.\n *\n * @param startIndex the beginning (inclusive) of the subrange to decode, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to decode, size of this array by default.\n * @param throwOnInvalidSequence specifies whether to throw an exception on malformed byte sequence or replace it by the replacement char `\\uFFFD`.\n *\n * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].\n * @throws CharacterCodingException if the byte array contains malformed UTF-8 byte sequence and [throwOnInvalidSequence] is true.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun ByteArray.decodeToString(\n   startIndex: Int = 0,\n   endIndex: Int = this.size,\n   throwOnInvalidSequence: Boolean = false\n): String {\n   AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)\n   return decodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)\n}\n\n/**\n * Encodes this string to an array of bytes in UTF-8 encoding.\n *\n * Any malformed char sequence is replaced by the replacement byte sequence.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun String.encodeToByteArray(): ByteArray {\n   return encodeUtf8(this, 0, length, false)\n}\n\n/**\n * Encodes this string or its substring to an array of bytes in UTF-8 encoding.\n *\n * @param startIndex the beginning (inclusive) of the substring to encode, 0 by default.\n * @param endIndex the end (exclusive) of the substring to encode, length of this string by default.\n * @param throwOnInvalidSequence specifies whether to throw an exception on malformed char sequence or replace.\n *\n * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the length of this string.\n * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].\n * @throws CharacterCodingException if this string contains malformed char sequence and [throwOnInvalidSequence] is true.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun String.encodeToByteArray(\n   startIndex: Int = 0,\n   endIndex: Int = this.length,\n   throwOnInvalidSequence: Boolean = false\n): ByteArray {\n   AbstractList.checkBoundsIndexes(startIndex, endIndex, length)\n   return encodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)\n}\n\n/**\n * Returns a copy of this string converted to upper case using the rules of the default locale.\n */\n@Deprecated(\"Use uppercase() instead.\", ReplaceWith(\"uppercase()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\n@kotlin.internal.InlineOnly\npublic actual inline fun String.toUpperCase(): String = asDynamic().toUpperCase()\n\n/**\n * Returns a copy of this string converted to upper case using Unicode mapping rules of the invariant locale.\n *\n * This function supports one-to-many and many-to-one character mapping,\n * thus the length of the returned string can be different from the length of the original string.\n *\n * @sample

samples.text.Strings.uppercase\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli
c actual inline fun String.uppercase(): String = asDynamic().toUpperCase()\n\n/**\n * Returns a copy of this string
converted to lower case using the rules of the default locale.\n */\n@Deprecated(\"Use lowercase() instead.\",
ReplaceWith(\"lowercase()\"))\n@DeprecatedSinceKotlin(warningSince =
\"1.5\")\n@kotlin.internal.InlineOnly\npublic actual inline fun String.toLowerCase(): String =
asDynamic().toLowerCase()\n\n/**\n * Returns a copy of this string converted to lower case using Unicode
mapping rules of the invariant locale.\n *\n * This function supports one-to-many and many-to-one character
mapping,\n * thus the length of the returned string can be different from the length of the original string.\n *\n *
@sample samples.text.Strings.lowercase\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli
c actual inline fun String.lowercase(): String = asDynamic().toLowerCase()\n\n@kotlin.internal.InlineOnly\ninternal
actual inline fun String.nativeIndexOf(str: String, fromIndex: Int): Int = asDynamic().indexOf(str,
fromIndex)\n\n@kotlin.internal.InlineOnly\ninternal actual inline fun String.nativeLastIndexOf(str: String,
fromIndex: Int): Int = asDynamic().lastIndexOf(str, fromIndex)\n\n@kotlin.internal.InlineOnly\ninternal inline fun
String.nativeStartsWith(s: String, position: Int): Boolean = asDynamic().startsWith(s,
position)\n\n@kotlin.internal.InlineOnly\ninternal inline fun String.nativeEndsWith(s: String): Boolean =
asDynamic().endsWith(s)\n\n@kotlin.internal.InlineOnly\npublic actual inline fun String.substring(startIndex: Int):
String = asDynamic().substring(startIndex)\n\n@kotlin.internal.InlineOnly\npublic actual inline fun
String.substring(startIndex: Int, endIndex: Int): String = asDynamic().substring(startIndex,
endIndex)\n\n@Deprecated(\"Use String.plus() instead\", ReplaceWith(\"this +
str\"))\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun
String.concat(str: String): String = asDynamic().concat(str)\n\n@Deprecated(\"Use Regex.findAll() instead or
invoke matches() on String dynamically:
this.asDynamic().match(regex)\")\n@DeprecatedSinceKotlin(warningSince =
\"1.6\")\n@kotlin.internal.InlineOnly\npublic inline fun String.match(regex: String): Array<String>? =
asDynamic().match(regex)\n\n//native public fun String.trim(): String\n//TODO: String.replace to implement
effective trimLeading and trimTrailing\n\n@kotlin.internal.InlineOnly\ninternal inline fun
String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern,
replacement)\n\n/**\n * Compares two strings lexicographically, optionally ignoring case differences.\n *\n * If
[ignoreCase] is true, the result of `Char.uppercaseChar().lowercaseChar()` on each character is compared.\n
*/\n@SinceKotlin(\"1.2\")\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic
actual fun String.compareTo(other: String, ignoreCase: Boolean = false): Int {\n if (ignoreCase) {\n val n1 =
this.length\n val n2 = other.length\n val min = minOf(n1, n2)\n if (min == 0) return n1 - n2\n for
(index in 0 until min) {\n var thisChar = this[index]\n var otherChar = other[index]\n\n if
(thisChar != otherChar) {\n thisChar = thisChar.uppercaseChar()\n otherChar =
otherChar.uppercaseChar()\n\n if (thisChar != otherChar) {\n thisChar =
thisChar.lowercaseChar()\n otherChar = otherChar.lowercaseChar()\n\n if (thisChar !=
otherChar) {\n return thisChar.compareTo(otherChar)\n }\n }\n }\n
}\n return n1 - n2\n } else {\n return compareTo(other)\n }\n}\n\n/**\n * Returns `true` if the contents
of this char sequence are equal to the contents of the specified [other],\n * i.e. both char sequences contain the same
number of the same characters in the same order.\n *\n * @sample samples.text.Strings.contentEquals\n
*/\n@SinceKotlin(\"1.5\")\npublic actual infix fun CharSequence?.contentEquals(other: CharSequence?): Boolean =
contentEqualsImpl(other)\n\n/**\n * Returns `true` if the contents of this char sequence are equal to the contents of
the specified [other], optionally ignoring case difference.\n *\n * @param ignoreCase `true` to ignore character case
when comparing contents.\n *\n * @sample samples.text.Strings.contentEquals\n */\n@SinceKotlin(\"1.5\")\npublic
actual fun CharSequence?.contentEquals(other: CharSequence?, ignoreCase: Boolean): Boolean {\n return if
(ignoreCase)\n this.contentEqualsIgnoreCaseImpl(other)\n else\n

this.contentEqualsImpl(other)\n}\n\n\nprivate val STRING_CASE_INSENSITIVE_ORDER = Comparator<String> { a, b -> a.compareTo(b, ignoreCase = true) }\n\n@SinceKotlin(\"1.2\")\npublic actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String>\n get() = STRING_CASE_INSENSITIVE_ORDER\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"CharsKt\")\n\npackage kotlin.text\n\n/**\n * Returns the numeric value of the decimal digit that this Char represents.\n * Throws an exception if this Char is not a valid decimal digit.\n *\n * A Char is considered to represent a decimal digit if [isDigit] is true for the Char.\n * In this case, the Unicode decimal digit value of the character is returned.\n *\n * @sample samples.text.Chars.digitToInt\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun Char.digitToInt(): Int {\n return digitOf(this, 10).also {\n if (it < 0) throw IllegalArgumentException(\"Char $this is not a decimal digit\")\n }\n}\n\n/**\n * Returns the numeric value of the digit that this Char represents in the specified [radix].\n * Throws an exception if the [radix] is not in the range `2..36` or if this Char is not a valid digit in the specified [radix].\n *\n * A Char is considered to represent a digit in the specified [radix] if at least one of the following is true:\n * - [isDigit] is `true` for the Char and the Unicode decimal digit value of the character is less than the specified [radix]. In this case the decimal digit value is returned.\n * - The Char is one of the uppercase Latin letters 'A' through 'Z' and its [code] is less than `radix + 'A'.code - 10`. In this case, `this.code - 'A'.code + 10` is returned.\n * - The Char is one of the lowercase Latin letters 'a' through 'z' and its [code] is less than `radix + 'a'.code - 10`. In this case, `this.code - 'a'.code + 10` is returned.\n * - The Char is one of the fullwidth Latin capital letters '\\uFF21' through '\\uFF3A' and its [code] is less than `radix + 0xFF21 - 10`. In this case, `this.code - 0xFF21 + 10` is returned.\n * - The Char is one of the fullwidth Latin small letters '\\uFF41' through '\\uFF5A' and its [code] is less than `radix + 0xFF41 - 10`. In this case, `this.code - 0xFF41 + 10` is returned.\n *\n * @sample samples.text.Chars.digitToInt\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun Char.digitToInt(radix: Int): Int {\n return digitToIntOrNull(radix) ?: throw IllegalArgumentException(\"Char $this is not a digit in the given radix=$radix\")\n}\n\n/**\n *\n * Returns the numeric value of the decimal digit that this Char represents, or `null` if this Char is not a valid decimal digit.\n *\n * A Char is considered to represent a decimal digit if [isDigit] is true for the Char.\n * In this case, the Unicode decimal digit value of the character is returned.\n *\n * @sample samples.text.Chars.digitToIntOrNull\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun Char.digitToIntOrNull(): Int? {\n return digitOf(this, 10).takeIf { it >= 0 }\n}\n\n/**\n * Returns the numeric value of the digit that this Char represents in the specified [radix], or `null` if this Char is not a valid digit in the specified [radix].\n * Throws an exception if the [radix] is not in the range `2..36`.\n *\n * A Char is considered to represent a digit in the specified [radix] if at least one of the following is true:\n * - [isDigit] is `true` for the Char and the Unicode decimal digit value of the character is less than the specified [radix]. In this case the decimal digit value is returned.\n * - The Char is one of the uppercase Latin letters 'A' through 'Z' and its [code] is less than `radix + 'A'.code - 10`. In this case, `this.code - 'A'.code + 10` is returned.\n * - The Char is one of the lowercase Latin letters 'a' through 'z' and its [code] is less than `radix + 'a'.code - 10`. In this case, `this.code - 'a'.code + 10` is returned.\n * - The Char is one of the fullwidth Latin capital letters '\\uFF21' through '\\uFF3A' and its [code] is less than `radix + 0xFF21 - 10`. In this case, `this.code - 0xFF21 + 10` is returned.\n * - The Char is one of the fullwidth Latin small letters '\\uFF41' through '\\uFF5A' and its [code] is less than `radix + 0xFF41 - 10`. In this case, `this.code - 0xFF41 + 10` is returned.\n *\n * @sample samples.text.Chars.digitToIntOrNull\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun Char.digitToIntOrNull(radix: Int): Int? {\n checkRadix(radix)\n return digitOf(this, radix).takeIf { it >= 0 }\n}\n\n/**\n * Returns the Char that represents this decimal digit.\n * Throws an exception if this value is not in the range `0..9`.\n *\n * If this value is in `0..9`, the decimal digit Char with code `'0'.code + this` is returned.\n *\n *

@sample samples.text.Chars.digitToChar\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun Int.digitToChar(): Char {\n   if (this in 0..9) {\n   return '0' + this\n   }\n   throw IllegalArgumentException(\"Int $this is not a decimal digit\")\n}\n\n/**\n * Returns the Char that represents this numeric digit value in the specified [radix].\n * Throws an exception if the [radix] is not in the range `2..36` or if this value is not in the range `0 until radix`.\n *\n * If this value is less than `10`, the decimal digit Char with code `'0'.code + this` is returned.\n * Otherwise, the uppercase Latin letter with code `'A'.code + this - 10` is returned.\n *\n * @sample samples.text.Chars.digitToChar\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun Int.digitToChar(radix: Int): Char {\n   if (radix !in 2..36) {\n   throw IllegalArgumentException(\"Invalid radix: $radix. Valid radix values are in range 2..36\")\n   }\n   if (this < 0 || this >= radix) {\n   throw IllegalArgumentException(\"Digit $this does not represent a valid digit in radix $radix\")\n   }\n   return if (this < 10) {\n   '0' + this\n   } else {\n   'A' + this - 10\n   }\n}\n\n/**\n * Converts this character to lower case using Unicode mapping rules of the invariant locale.\n */\n@Deprecated(\"Use lowercaseChar() instead.\", ReplaceWith(\"lowercaseChar()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic expect fun Char.toLowerCase(): Char\n\n/**\n * Converts this character to lower case using Unicode mapping rules of the invariant locale.\n *\n * This function performs one-to-one character mapping.\n * To support one-to-many character mapping use the [lowercase] function.\n * If this character has no mapping equivalent, the character itself is returned.\n *\n * @sample samples.text.Chars.lowercase\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Char.lowercaseChar(): Char\n\n/**\n * Converts this character to lower case using Unicode mapping rules of the invariant locale.\n *\n * This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.\n * For example, `\"\\u0130'.lowercase()` returns `\"\\u0069\\u0307\"`,\n * where `\"\\u0130'` is the LATIN CAPITAL LETTER I WITH DOT ABOVE character (`\ufffd\ufffd`).\n * If this character has no lower case mapping, the result of `toString()` of this char is returned.\n *\n * @sample samples.text.Chars.lowercase\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Char.lowercase(): String\n\n/**\n * Converts this character to upper case using Unicode mapping rules of the invariant locale.\n */\n@Deprecated(\"Use uppercaseChar() instead.\", ReplaceWith(\"uppercaseChar()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic expect fun Char.toUpperCase(): Char\n\n/**\n * Converts this character to upper case using Unicode mapping rules of the invariant locale.\n *\n * This function performs one-to-one character mapping.\n * To support one-to-many character mapping use the [uppercase] function.\n * If this character has no mapping equivalent, the character itself is returned.\n *\n * @sample samples.text.Chars.uppercase\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Char.uppercaseChar(): Char\n\n/**\n * Converts this character to upper case using Unicode mapping rules of the invariant locale.\n *\n * This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.\n * For example, `\"\\uFB00'.uppercase()` returns `\"\\u0046\\u0046\"`,\n * where `\"\\uFB00'` is the LATIN SMALL LIGATURE FF character (`\ufffd\ufffd\ufffd`).\n * If this character has no upper case mapping, the result of `toString()` of this char is returned.\n *\n * @sample samples.text.Chars.uppercase\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Char.uppercase(): String\n\n/**\n * Converts this character to title case using Unicode mapping rules of the invariant locale.\n *\n * This function performs one-to-one character mapping.\n * To support one-to-many character mapping use the [titlecase] function.\n * If this character has no mapping equivalent, the result of calling [uppercaseChar] is returned.\n *\n * @sample samples.text.Chars.titlecase\n */\n@SinceKotlin(\"1.5\")\npublic expect fun Char.titlecaseChar(): Char\n\n/**\n * Converts this character to title case using Unicode mapping rules of the invariant locale.\n *\n * This function supports one-to-many character mapping, thus the length of the returned string can be greater than one.\n * For example, `\"\\uFB00'.titlecase()` returns `\"\\u0046\\u0066\"`,\n * where `\"\\uFB00'` is the LATIN SMALL LIGATURE FF character (`\ufffd\ufffd\ufffd`).\n * If this character has no title

case mapping, the result of [uppercase] is returned instead.\n *\n * @sample samples.text.Chars.titlecase\n */\n@SinceKotlin(\"1.5\")\npublic fun Char.titlecase(): String = titlecaseImpl()\n\n/**\n * Concatenates this Char and a String.\n *\n * @sample samples.text.Chars.plus\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun Char.plus(other: String): String = this.toString() + other\n\n/**\n * Returns `true` if this character is equal to the [other] character, optionally ignoring character case.\n *\n * Two characters are considered equal ignoring case if `Char.uppercaseChar().lowercaseChar()` on each character produces the same result.\n *\n * @param ignoreCase `true` to ignore character case when comparing characters. By default `false`.\n * @sample samples.text.Chars.equals\n */\npublic fun Char.equals(other: Char, ignoreCase: Boolean = false): Boolean {\n   if (this == other) return true\n   if (!ignoreCase) return false\n\n   val thisUpper = this.uppercaseChar()\n   val otherUpper = other.uppercaseChar()\n\n   return thisUpper == otherUpper || thisUpper.lowercaseChar() == otherUpper.lowercaseChar()\n}\n\n/**\n * Returns `true` if this character is a Unicode surrogate code unit.\n */\npublic fun Char.isSurrogate(): Boolean = this in Char.MIN_SURROGATE..Char.MAX_SURROGATE\n\n/**\n * Returns the Unicode general category of this character.\n */\n@SinceKotlin(\"1.5\")\npublic expect val Char.category: CharCategory\n\n/**\n * Returns `true` if this character (Unicode code point) is defined in Unicode.\n *\n * A character is considered to be defined in Unicode if its [category] is not [CharCategory.UNASSIGNED].\n */\n@SinceKotlin(\"1.5\")\npublic expect fun Char.isDefined(): Boolean\n\n/**\n * Returns `true` if this character is a letter.\n *\n * A character is considered to be a letter if its [category] is [CharCategory.UPPERCASE_LETTER],\n * [CharCategory.LOWERCASE_LETTER], [CharCategory.TITLECASE_LETTER], [CharCategory.MODIFIER_LETTER], or [CharCategory.OTHER_LETTER].\n *\n * @sample samples.text.Chars.isLetter\n */\n@SinceKotlin(\"1.5\")\npublic expect fun Char.isLetter(): Boolean\n\n/**\n * Returns `true` if this character is a letter or digit.\n *\n * @see isLetter\n * @see isDigit\n *\n * @sample samples.text.Chars.isLetterOrDigit\n */\n@SinceKotlin(\"1.5\")\npublic expect fun Char.isLetterOrDigit(): Boolean\n\n/**\n * Returns `true` if this character is a digit.\n *\n * A character is considered to be a digit if its [category] is [CharCategory.DECIMAL_DIGIT_NUMBER].\n *\n * @sample samples.text.Chars.isDigit\n */\n@SinceKotlin(\"1.5\")\npublic expect fun Char.isDigit(): Boolean\n\n/**\n * Returns `true` if this character is upper case.\n *\n * A character is considered to be an upper case character if its [category] is [CharCategory.UPPERCASE_LETTER],\n * or it has contributory property `Other_Uppercase` as defined by the Unicode Standard.\n *\n * @sample samples.text.Chars.isUpperCase\n */\n@SinceKotlin(\"1.5\")\npublic expect fun Char.isUpperCase(): Boolean\n\n/**\n * Returns `true` if this character is lower case.\n *\n * A character is considered to be a lower case character if its [category] is [CharCategory.LOWERCASE_LETTER],\n * or it has contributory property `Other_Lowercase` as defined by the Unicode Standard.\n *\n * @sample samples.text.Chars.isLowerCase\n */\n@SinceKotlin(\"1.5\")\npublic expect fun Char.isLowerCase(): Boolean\n\n/**\n * Returns `true` if this character is a title case letter.\n *\n * A character is considered to be a title case letter if its [category] is [CharCategory.TITLECASE_LETTER].\n *\n * @sample samples.text.Chars.isTitleCase\n */\n@SinceKotlin(\"1.5\")\npublic expect fun Char.isTitleCase(): Boolean\n\n/**\n * Returns `true` if this character is an ISO control character.\n *\n * A character is considered to be an ISO control character if its [category] is [CharCategory.CONTROL],\n * meaning the Char is in the range `'\\u0000'..'\\u001F'` or in the range `'\\u007F'..'\\u009F'`.\n *\n * @sample samples.text.Chars.isISOControl\n */\n@SinceKotlin(\"1.5\")\npublic expect fun Char.isISOControl(): Boolean\n\n/**\n * Determines whether a character is whitespace according to the Unicode standard.\n * Returns `true` if the character is whitespace.\n *\n * @sample samples.text.Chars.isWhitespace\n */\npublic expect fun Char.isWhitespace(): Boolean\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\n\n/**\n * Creates a Char with the specified [code], or throws an exception if the [code] is out of `Char.MIN_VALUE.code..Char.MAX_VALUE.code`.\n *\n * If the program that calls this function is written in a way that only valid [code] is passed as the argument,\n * using the overload that takes a [UShort] argument is preferable (`Char(intValue.toUShort())`).\n * That overload doesn't check validity of the argument, and may

improve program performance when the function is called routinely inside a loop.\n *\n * @sample samples.text.Chars.charFromCode\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun Char(code: Int): Char {\n   if (code < Char.MIN_VALUE.code || code > Char.MAX_VALUE.code) {\n   throw IllegalArgumentException(\"Invalid Char code: $code\")\n   }\n   return code.toChar()\n}\n\n/**\n * Creates a Char with the specified [code].\n *\n * @sample samples.text.Chars.charFromCode\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"NO_ACTUAL_FOR_EXPECT\")\npublic expect fun Char(code: UShort): Char\n\n/**\n * Returns the code of this Char.\n *\n * Code of a Char is the value it was constructed with, and the UTF-16 code unit corresponding to this Char.\n *\n * @sample samples.text.Chars.code\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\n@Suppress(\"DEPRECATION\")\npublic inline val Char.code: Int get() = this.toInt()\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"SequencesKt\")\n\npackage kotlin.sequences\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\n\n/**\n * Returns `true` if [element] is found in the sequence.\n *\n * The operation is _terminal_.\n */\npublic operator fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {\n   return indexOf(element) >= 0\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this sequence.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic fun <T> Sequence<T>.elementAt(index: Int): T {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"Sequence doesn't contain element at index $index.\") }\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this sequence.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\npublic fun <T> Sequence<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {\n   if (index < 0)\n   return defaultValue(index)\n   val iterator = iterator()\n   var count = 0\n   while (iterator.hasNext()) {\n   val element = iterator.next()\n   if (index == count++)\n   return element\n   }\n   return defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this sequence.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\npublic fun <T> Sequence<T>.elementAtOrNull(index: Int): T? {\n   if (index < 0)\n   return null\n   val iterator = iterator()\n   var count = 0\n   while (iterator.hasNext()) {\n   val element = iterator.next()\n   if (index == count++)\n   return element\n   }\n   return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.find(predicate: (T) -> Boolean): T? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.findLast(predicate: (T) -> Boolean): T? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the sequence is empty.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.first(): T {\n   val iterator = iterator()\n   if (!iterator.hasNext())\n   throw NoSuchElementException(\"Sequence is empty.\")\n   return iterator.next()\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.first(predicate: (T) -> Boolean): T {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Sequence contains no element matching the predicate.\")\n}\n\n/**\n

```
* Returns the first non-null value produced by [transform] function being applied to elements of this sequence in iteration order,\n * or throws [NoSuchElementException] if no non-null value was produced.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.firstNotNullOf\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Any> Sequence<T>.firstNotNullOf(transform: (T) -> R?): R {\n    return firstNotNullOfOrNull(transform) ?: throw NoSuchElementException(\"No element of the sequence was transformed to a non-null value.\")\n}\n\n/**\n * Returns the first non-null value produced by [transform] function being applied to elements of this sequence in iteration order,\n * or `null` if no non-null value was produced.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.firstNotNullOf\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Any> Sequence<T>.firstNotNullOfOrNull(transform: (T) -> R?): R? {\n    for (element in this) {\n        val result = transform(element)\n        if (result != null) {\n            return result\n        }\n    }\n    return null\n}\n\n/**\n * Returns the first element, or `null` if the sequence is empty.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.firstOrNull(): T? {\n    val iterator = iterator()\n    if (!iterator.hasNext())\n        return null\n    return iterator.next()\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.firstOrNull(predicate: (T) -> Boolean): T? {\n    for (element in this) if (predicate(element)) return element\n    return null\n}\n\n/**\n * Returns first index of [element], or -1 if the sequence does not contain element.\n *\n * The operation is _terminal_.\n */\npublic fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOf(element: T): Int {\n    var index = 0\n    for (item in this) {\n        checkIndexOverflow(index)\n        if (element == item)\n            return index\n        index++\n    }\n    return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the sequence does not contain such element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.indexOfFirst(predicate: (T) -> Boolean): Int {\n    var index = 0\n    for (item in this) {\n        checkIndexOverflow(index)\n        if (predicate(item))\n            return index\n        index++\n    }\n    return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the sequence does not contain such element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.indexOfLast(predicate: (T) -> Boolean): Int {\n    var lastIndex = -1\n    var index = 0\n    for (item in this) {\n        checkIndexOverflow(index)\n        if (predicate(item))\n            lastIndex = index\n        index++\n    }\n    return lastIndex\n}\n\n/**\n * Returns the last element.\n *\n * The operation is _terminal_.\n * \n * @throws NoSuchElementException if the sequence is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun <T> Sequence<T>.last(): T {\n    val iterator = iterator()\n    if (!iterator.hasNext())\n        throw NoSuchElementException(\"Sequence is empty.\")\n    var last = iterator.next()\n    while (iterator.hasNext())\n        last = iterator.next()\n    return last\n}\n\n/**\n * Returns the last element matching the given [predicate].\n *\n * The operation is _terminal_.\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun <T> Sequence<T>.last(predicate: (T) -> Boolean): T {\n    var last: T? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            last = element\n            found = true\n        }\n    }\n    if (!found) throw NoSuchElementException(\"Sequence contains no element matching the predicate.\")\n    @Suppress(\"UNCHECKED_CAST\")\n    return last as T\n}\n\n/**\n * Returns last index of [element], or -1 if the sequence does not contain element.\n *\n * The operation is _terminal_.\n */\npublic fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.lastIndexOf(element: T): Int {\n    var lastIndex = -1\n    var index = 0\n    for (item in this) {\n        checkIndexOverflow(index)\n        if (element == item)\n            lastIndex = index\n        index++\n    }\n    return lastIndex\n}\n\n/**\n * Returns the last element, or `null` if the sequence is empty.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic fun <T> Sequence<T>.lastOrNull(): T? {\n    val iterator = iterator()\n    if (!iterator.hasNext())\n        return null\n    var last = iterator.next()\n    while (iterator.hasNext())\n        last = iterator.next()\n    return last\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n
```

*\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Elements.last\n */\npublic inline fun <T> Sequence<T>.lastOrNull(predicate: (T) -> Boolean): T? {\n var last: T? = null\n for (element in this) {\n if (predicate(element)) {\n last = element\n }\n }\n return last\n}\n\n/**\n * Returns the single element, or throws an exception if the sequence is empty or has more than one element.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.single(): T {\n val iterator = iterator()\n if (!iterator.hasNext())\n throw NoSuchElementException(\"Sequence is empty.\")\n val single = iterator.next()\n if (iterator.hasNext())\n throw IllegalArgumentException(\"Sequence has more than one element.\")\n return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.single(predicate: (T) -> Boolean): T {\n var single: T? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Sequence contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Sequence contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as T\n}\n\n/**\n * Returns single element, or `null` if the sequence is empty or has more than one element.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.singleOrNull(): T? {\n val iterator = iterator()\n if (!iterator.hasNext())\n return null\n val single = iterator.next()\n if (iterator.hasNext())\n return null\n return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.singleOrNull(predicate: (T) -> Boolean): T? {\n var single: T? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) return null\n single = element\n found = true\n }\n }\n if (!found) return null\n return single\n}\n\n/**\n * Returns a sequence containing all elements except first [n] elements.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun <T> Sequence<T>.drop(n: Int): Sequence<T> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return when {\n n == 0 -> this\n this is DropTakeSequence -> this.drop(n)\n else -> DropSequence(this, n)\n }\n}\n\n/**\n * Returns a sequence containing all elements except first elements that satisfy the given [predicate].\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun <T> Sequence<T>.dropWhile(predicate: (T) -> Boolean): Sequence<T> {\n return DropWhileSequence(this, predicate)\n}\n\n/**\n * Returns a sequence containing only elements matching the given [predicate].\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic fun <T> Sequence<T>.filter(predicate: (T) -> Boolean): Sequence<T> {\n return FilteringSequence(this, true, predicate)\n}\n\n/**\n * Returns a sequence containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic fun <T> Sequence<T>.filterIndexed(predicate: (index: Int, T) -> Boolean): Sequence<T> {\n // TODO: Rewrite with generalized MapFilterIndexingSequence\n return TransformingSequence(FilteringSequence(IndexingSequence(this), true, { predicate(it.index, it.value) }), { it.value })\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <T, C : MutableCollection<in T>> Sequence<T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {\n forEachIndexed { index, element ->\n if (predicate(index, element)) destination.add(element)\n }\n return destination\n}\n\n/**\n * Returns a sequence containing all elements that are instances of specified type parameter R.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Filtering.filterIsInstance\n */\npublic inline fun <reified R>

Sequence<*>.filterIsInstance(): Sequence<@kotlin.internal.NoInfer R> {\n @Suppress(\"UNCHECKED_CAST\")\n return filter { it is R } as Sequence<R>\n}\n\n/**\n * Appends all elements that are instances of specified type parameter R to the given [destination].\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Filtering.filterIsInstanceTo\n */\npublic inline fun <reified R, C : MutableCollection<in R>> Sequence<*>.filterIsInstanceTo(destination: C): C {\n for (element in this) if (element is R) destination.add(element)\n return destination\n}\n\n/**\n * Returns a sequence containing all elements not matching the given [predicate].\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic fun <T> Sequence<T>.filterNot(predicate: (T) -> Boolean): Sequence<T> {\n return FilteringSequence(this, false, predicate)\n}\n\n/**\n * Returns a sequence containing all elements that are not `null`.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Filtering.filterNotNull\n */\npublic fun <T : Any> Sequence<T?>.filterNotNull(): Sequence<T> {\n @Suppress(\"UNCHECKED_CAST\")\n return filterNot { it == null } as Sequence<T>\n}\n\n/**\n * Appends all elements that are not `null` to the given [destination].\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Filtering.filterNotNullTo\n */\npublic fun <C : MutableCollection<in T>, T : Any> Sequence<T?>.filterNotNullTo(destination: C): C {\n for (element in this) if (element != null) destination.add(element)\n return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <T, C : MutableCollection<in T>> Sequence<T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {\n for (element in this) if (!predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <T, C : MutableCollection<in T>> Sequence<T>.filterTo(destination: C, predicate: (T) -> Boolean): C {\n for (element in this) if (predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Returns a sequence containing first [n] elements.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic fun <T> Sequence<T>.take(n: Int): Sequence<T> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return when {\n n == 0 -> emptySequence()\n this is DropTakeSequence -> this.take(n)\n else -> TakeSequence(this, n)\n }\n}\n\n/**\n * Returns a sequence containing first elements satisfying the given [predicate].\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Transformations.takeWhile\n */\npublic fun <T> Sequence<T>.takeWhile(predicate: (T) -> Boolean): Sequence<T> {\n return TakeWhileSequence(this, predicate)\n}\n\n/**\n * Returns a sequence that yields elements of this sequence sorted according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n *\n * The operation is _intermediate_ and _stateful_.\n */\npublic fun <T : Comparable<T>> Sequence<T>.sorted(): Sequence<T> {\n return object : Sequence<T> {\n override fun iterator(): Iterator<T> {\n val sortedList = this@sorted.toMutableList()\n sortedList.sort()\n return sortedList.iterator()\n }\n }\n}\n\n/**\n * Returns a sequence that yields elements of this sequence sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n *\n * The operation is _intermediate_ and _stateful_.\n * \n * @sample samples.collections.Collections.Sorting.sortedBy\n */\npublic inline fun <T, R : Comparable<R>> Sequence<T>.sortedBy(crossinline selector: (T) -> R?): Sequence<T> {\n return sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a sequence that yields elements of this sequence sorted descending according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n *\n * The operation is _intermediate_ and _stateful_.\n */\npublic inline fun <T, R : Comparable<R>> Sequence<T>.sortedByDescending(crossinline selector: (T) -> R?): Sequence<T> {\n return sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a sequence that yields elements of this sequence sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements

preserve their order relative to each other after sorting.\n *\n * The operation is _intermediate_ and _stateful_.\n */\npublic fun <T : Comparable<T>> Sequence<T>.sortedDescending(): Sequence<T> {\n return sortedWith(reverseOrder())\n}\n\n/**\n * Returns a sequence that yields elements of this sequence sorted according to the specified [comparator].\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n *\n * The operation is _intermediate_ and _stateful_.\n */\npublic fun <T> Sequence<T>.sortedWith(comparator: Comparator<in T>): Sequence<T> {\n return object : Sequence<T> {\n override fun iterator(): Iterator<T> {\n val sortedList = this@sortedWith.toMutableList()\n sortedList.sortWith(comparator)\n return sortedList.iterator()\n }\n }\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given sequence.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original sequence.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.associate\n */\npublic inline fun <T, K, V> Sequence<T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {\n return associateTo(LinkedHashMap<K, V>(), transform)\n}\n\n/**\n * Returns a [Map] containing the elements from the given sequence indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original sequence.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.associateBy\n */\npublic inline fun <T, K> Sequence<T>.associateBy(keySelector: (T) -> K): Map<K, T> {\n return associateByTo(LinkedHashMap<K, T>(), keySelector)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given sequence.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original sequence.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.associateByWithValueTransform\n */\npublic inline fun <T, K, V> Sequence<T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {\n return associateByTo(LinkedHashMap<K, V>(), keySelector, valueTransform)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given sequence\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.associateByTo\n */\npublic inline fun <T, K, M : MutableMap<in K, in T>> Sequence<T>.associateByTo(destination: M, keySelector: (T) -> K): M {\n for (element in this) {\n destination.put(keySelector(element), element)\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given sequence.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.associateByToWithValueTransform\n */\npublic inline fun <T, K, V, M : MutableMap<in K, in V>> Sequence<T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {\n for (element in this) {\n destination.put(keySelector(element), valueTransform(element))\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given sequence.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.associateTo\n */\npublic inline fun <T, K, V, M : MutableMap<in K, in V>> Sequence<T>.associateTo(destination: M, transform: (T) -> Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n return destination\n}\n\n/**\n * Returns a [Map] where keys are elements from the given sequence and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original sequence.\n *\n * The

operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.3\")\npublic inline fun <K, V> Sequence<K>.associateWith(valueSelector: (K) -> V): Map<K, V> {\n   val result = LinkedHashMap<K, V>()\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given sequence,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.3\")\npublic inline fun <K, V, M : MutableMap<in K, in V>> Sequence<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {\n   for (element in this) {\n   destination.put(element, valueSelector(element))\n   }\n   return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n *\n * The operation is _terminal_.\n */\npublic fun <T, C : MutableCollection<in T>> Sequence<T>.toCollection(destination: C): C {\n   for (item in this) {\n   destination.add(item)\n   }\n   return destination\n}\n\n/**\n * Returns a new [HashSet] of all elements.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.toHashSet(): HashSet<T> {\n   return toCollection(HashSet<T>())\n}\n\n/**\n * Returns a [List] containing all elements.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.toList(): List<T> {\n   return this.toMutableList().optimizeReadOnlyList()\n}\n\n/**\n * Returns a new [MutableList] filled with all elements of this sequence.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.toMutableList(): MutableList<T> {\n   return toCollection(ArrayList<T>())\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original sequence.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.toSet(): Set<T> {\n   return toCollection(LinkedHashSet<T>()).optimizeReadOnlySet()\n}\n\n/**\n * Returns a single sequence of all elements from results of [transform] function being invoked on each element of original sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIterable\")\npublic fun <T, R> Sequence<T>.flatMap(transform: (T) -> Iterable<R>): Sequence<R> {\n   return FlatteningSequence(this, transform, Iterable<R>::iterator)\n}\n\n/**\n * Returns a single sequence of all elements from results of [transform] function being invoked on each element of original sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic fun <T, R> Sequence<T>.flatMap(transform: (T) -> Sequence<R>): Sequence<R> {\n   return FlatteningSequence(this, transform, Sequence<R>::iterator)\n}\n\n/**\n * Returns a single sequence of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\npublic fun <T, R> Sequence<T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): Sequence<R> {\n   return flatMapIndexed(this, transform, Iterable<R>::iterator)\n}\n\n/**\n * Returns a single sequence of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedSequence\")\npublic fun <T, R> Sequence<T>.flatMapIndexed(transform: (index: Int, T) -> Sequence<R>): Sequence<R> {\n   return flatMapIndexed(this, transform, Sequence<R>::iterator)\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original sequence, to the given [destination].\n *\n * The operation is _terminal_.\n

*/\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Iterable<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(checkIndexOverflow(index++), element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original sequence, to the given [destination].\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedSequenceTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Sequence<R>): C {\n var index = 0\n for (element in this) {\n val list = transform(checkIndexOverflow(index++), element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original sequence, to the given [destination].\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIterableTo\")\npublic inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original sequence, to the given [destination].\n *\n * The operation is _terminal_.\n */\npublic inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return destination\n}\n\n/**\n * Groups elements of the original sequence by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original sequence.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <T, K> Sequence<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {\n return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original sequence\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original sequence.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <T, K, V> Sequence<T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {\n return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups elements of the original sequence by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<T>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original sequence\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K,

valueTransform: (T) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Creates a [Grouping] source from a sequence to be used later with one of group-and-fold operations\n * using the specified [keySelector] function to extract a key from each element.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Grouping.groupingByEachCount\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K> Sequence<T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {\n return object : Grouping<T, K> {\n override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()\n override fun keyOf(element: T): K = keySelector(element)\n }\n}\n\n/**\n * Returns a sequence containing the results of applying the given [transform] function\n * to each element in the original sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\npublic fun <T, R> Sequence<T>.map(transform: (T) -> R): Sequence<R> {\n return TransformingSequence(this, transform)\n}\n\n/**\n * Returns a sequence containing the results of applying the given [transform] function\n * to each element and its index in the original sequence.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic fun <T, R> Sequence<T>.mapIndexed(transform: (index: Int, T) -> R): Sequence<R> {\n return TransformingIndexedSequence(this, transform)\n}\n\n/**\n * Returns a sequence containing only the non-null results of applying the given [transform] function\n * to each element and its index in the original sequence.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic fun <T, R : Any> Sequence<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): Sequence<R> {\n return TransformingIndexedSequence(this, transform).filterNotNull()\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original sequence\n * and appends only the non-null results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T, R : Any, C : MutableCollection<in R>> Sequence<T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {\n forEachIndexed { index, element -> transform(index, element)?.let { destination.add(it) } }\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original sequence\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {\n var index = 0\n for (item in this)\n destination.add(transform(checkIndexOverflow(index++), item))\n return destination\n}\n\n/**\n * Returns a sequence containing only the non-null results of applying the given [transform] function\n * to each element in the original sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Transformations.mapNotNull\n */\npublic fun <T, R : Any> Sequence<T>.mapNotNull(transform: (T) -> R?): Sequence<R> {\n return TransformingSequence(this, transform).filterNotNull()\n}\n\n/**\n * Applies the given [transform] function to each element in the original sequence\n * and appends only the non-null results to the given [destination].\n *\n * The operation is _terminal_.\n */\npublic inline fun <T, R : Any, C : MutableCollection<in R>> Sequence<T>.mapNotNullTo(destination: C, transform: (T) -> R?): C {\n forEach { element -> transform(element)?.let { destination.add(it) } }\n return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original sequence\n * and appends the results to the given [destination].\n *\n * The operation is _terminal_.\n */\npublic inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.mapTo(destination: C, transform: (T) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return destination\n}\n\n/**\n * Returns a sequence that wraps each element of the original sequence\n * into an [IndexedValue] containing the index of that element and the element itself.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic fun <T> Sequence<T>.withIndex(): Sequence<IndexedValue<T>> {\n return IndexingSequence(this)\n}\n\n/**\n * Returns a sequence containing

only distinct elements from the given sequence.\n * \n * Among equal elements of the given sequence, only the first one will be present in the resulting sequence.\n * The elements in the resulting sequence are in the same order as they were in the source sequence.\n *\n * The operation is _intermediate_ and _stateful_.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun <T> Sequence<T>.distinct(): Sequence<T> {\n   return this.distinctBy { it }\n}\n\n/**\n * Returns a sequence containing only elements from the given sequence\n * having distinct keys returned by the given [selector] function.\n * \n * Among elements of the given sequence with equal keys, only the first one will be present in the resulting sequence.\n * The elements in the resulting sequence are in the same order as they were in the source sequence.\n *\n * The operation is _intermediate_ and _stateful_.\n * \n * @sample samples.collections.Collections.Transformations.distinctAndDistinctBy\n */\npublic fun <T, K> Sequence<T>.distinctBy(selector: (T) -> K): Sequence<T> {\n   return DistinctSequence(this, selector)\n}\n\n/**\n * Returns a new [MutableSet] containing all distinct elements from the given sequence.\n * \n * The returned set preserves the element iteration order of the original sequence.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.toMutableSet(): MutableSet<T> {\n   val set = LinkedHashSet<T>()\n   for (item in this) set.add(item)\n   return set\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun <T> Sequence<T>.all(predicate: (T) -> Boolean): Boolean {\n   for (element in this) if (!predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if sequence has at least one element.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun <T> Sequence<T>.any(): Boolean {\n   return iterator().hasNext()\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun <T> Sequence<T>.any(predicate: (T) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return true\n   return false\n}\n\n/**\n * Returns the number of elements in this sequence.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.count(): Int {\n   var count = 0\n   for (element in this) checkCountOverflow(++count)\n   return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.count(predicate: (T) -> Boolean): Int {\n   var count = 0\n   for (element in this) if (predicate(element)) checkCountOverflow(++count)\n   return count\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the sequence is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T, R> Sequence<T>.fold(initial: R, operation: (acc: R, T) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original sequence.\n * \n * Returns the specified [initial] value if the sequence is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T, R> Sequence<T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {\n   var index = 0\n   var accumulator = initial\n   for (element in this) accumulator = operation(checkIndexOverflow(index++), accumulator, element)\n   return accumulator\n}\n\n/**\n * Performs the given [action] on each element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.forEach(action: (T) -> Unit): Unit {\n   for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {\n   var index = 0\n   for (item in this) action(checkIndexOverflow(index++), item)\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\",

hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Sequence<Double>.max(): Double? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Sequence<Float>.max(): Float? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T : Comparable<T>> Sequence<T>.max(): T? {\n return maxOrNull()\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <T, R : Comparable<R>> Sequence<T>.maxBy(selector: (T) -> R): T? {\n return maxByOrNull(selector)\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, R : Comparable<R>> Sequence<T>.maxByOrNull(selector: (T) -> R): T? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var maxElem = iterator.next()\n if (!iterator.hasNext()) return maxElem\n var maxValue = selector(maxElem)\n do {\n val e = iterator.next()\n val v = selector(e)\n if (maxValue < v) {\n maxElem = e\n maxValue = v\n }\n } while (iterator.hasNext())\n return maxElem\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the sequence.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the sequence is empty.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.maxOf(selector: (T) -> Double): Double {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var maxValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the sequence.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the sequence is empty.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.maxOf(selector: (T) -> Float): Float {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var maxValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the sequence.\n * \n * @throws NoSuchElementException if the sequence is empty.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Sequence<T>.maxOf(selector: (T) -> R): R {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var maxValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n if (maxValue < v) {\n maxValue = v\n }\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the sequence or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.maxOfOrNull(selector: (T) -> Double): Double? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var maxValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n maxValue =

maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the sequence or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.maxOfOrNull(selector: (T) -> Float): Float? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the sequence or `null` if there are no elements.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Sequence<T>.maxOfOrNull(selector: (T) -> R): R? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the sequence.\n * \n * @throws NoSuchElementException if the sequence is empty.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Sequence<T>.maxOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {\n   val iterator = iterator()\n   if (!iterator.hasNext()) throw NoSuchElementException()\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the sequence or `null` if there are no elements.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Sequence<T>.maxOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var maxValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\npublic fun Sequence<Double>.maxOrNull(): Double? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var max = iterator.next()\n   while (iterator.hasNext()) {\n   val e = iterator.next()\n   max = maxOf(max, e)\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\npublic fun Sequence<Float>.maxOrNull(): Float? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var max = iterator.next()\n   while (iterator.hasNext()) {\n   val e = iterator.next()\n   max = maxOf(max, e)\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T : Comparable<T>> Sequence<T>.maxOrNull(): T? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var max = iterator.next()\n   while (iterator.hasNext()) {\n   val e = iterator.next()\n   if (max < e) max = e\n   }\n   return max\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T> Sequence<T>.maxWith(comparator: Comparator<in T>): T? {\n return maxWithOrNull(comparator)\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n *\n * The operation is _terminal_.\n

*/\n@SinceKotlin(\"1.4\")\npublic fun <T> Sequence<T>.maxWithOrNull(comparator: Comparator<in T>): T? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var max = iterator.next()\n while (iterator.hasNext()) {\n val e = iterator.next()\n if (comparator.compare(max, e) < 0) max = e\n }\n return max\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Sequence<Double>.min(): Double? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.1\")\npublic fun Sequence<Float>.min(): Float? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T : Comparable<T>> Sequence<T>.min(): T? {\n return minOrNull()\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <T, R : Comparable<R>> Sequence<T>.minBy(selector: (T) -> R): T? {\n return minByOrNull(selector)\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <T, R : Comparable<R>> Sequence<T>.minByOrNull(selector: (T) -> R): T? {\n val iterator = iterator()\n if (!iterator.hasNext()) return null\n var minElem = iterator.next()\n if (!iterator.hasNext()) return minElem\n var minValue = selector(minElem)\n do {\n val e = iterator.next()\n val v = selector(e)\n if (minValue > v) {\n minElem = e\n minValue = v\n }\n } while (iterator.hasNext())\n return minElem\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the sequence.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the sequence is empty.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.minOf(selector: (T) -> Double): Double {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the sequence.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the sequence is empty.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.minOf(selector: (T) -> Float): Float {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the sequence.\n * \n * @throws NoSuchElementException if the sequence is empty.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Sequence<T>.minOf(selector: (T) -> R): R {\n val iterator = iterator()\n if (!iterator.hasNext()) throw NoSuchElementException()\n var minValue = selector(iterator.next())\n while (iterator.hasNext()) {\n val v = selector(iterator.next())\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the sequence or `null` if there are no elements.\n * \n * If any of values produced by [selector]

function is `NaN`, the returned result is `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.minOfOrNull(selector: (T) -> Double): Double? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var minValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the sequence or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.minOfOrNull(selector: (T) -> Float): Float? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var minValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the sequence or `null` if there are no elements.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R : Comparable<R>> Sequence<T>.minOfOrNull(selector: (T) -> R): R? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var minValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the sequence.\n * \n * @throws NoSuchElementException if the sequence is empty.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Sequence<T>.minOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {\n   val iterator = iterator()\n   if (!iterator.hasNext()) throw NoSuchElementException()\n   var minValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the sequence or `null` if there are no elements.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T, R> Sequence<T>.minOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var minValue = selector(iterator.next())\n   while (iterator.hasNext()) {\n   val v = selector(iterator.next())\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\npublic fun Sequence<Double>.minOrNull(): Double? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var min = iterator.next()\n   while (iterator.hasNext()) {\n   val e = iterator.next()\n   min = minOf(min, e)\n   }\n   return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\npublic fun Sequence<Float>.minOrNull(): Float? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var min = iterator.next()\n   while (iterator.hasNext()) {\n   val e = iterator.next()\n   min = minOf(min, e)\n   }\n   return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T : Comparable<T>> Sequence<T>.minOrNull(): T? {\n   val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var min = iterator.next()\n   while (iterator.hasNext()) {\n   val e = iterator.next()\n   if (min > e) min

= e\n   }\n   return min\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun <T> Sequence<T>.minWith(comparator: Comparator<in T>): T? {\n return minWithOrNull(comparator)\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Sequence<T>.minWithOrNull(comparator: Comparator<in T>): T? {\n val iterator = iterator()\n   if (!iterator.hasNext()) return null\n   var min = iterator.next()\n   while (iterator.hasNext()) {\n   val e = iterator.next()\n   if (comparator.compare(min, e) > 0) min = e\n   }\n   return min\n}\n\n/**\n * Returns `true` if the sequence has no elements.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun <T> Sequence<T>.none(): Boolean {\n return !iterator().hasNext()\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun <T> Sequence<T>.none(predicate: (T) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns a sequence which performs the given [action] on each element of the original sequence as they pass through it.\n *\n * The operation is _intermediate_ and _stateless_.\n */\n@SinceKotlin(\"1.1\")\npublic fun <T> Sequence<T>.onEach(action: (T) -> Unit): Sequence<T> {\n   return map {\n   action(it)\n   it\n   }\n}\n\n/**\n * Returns a sequence which performs the given [action] on each element of the original sequence as they pass through it.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n *\n * The operation is _intermediate_ and _stateless_.\n */\n@SinceKotlin(\"1.4\")\npublic fun <T> Sequence<T>.onEachIndexed(action: (index: Int, T) -> Unit): Sequence<T> {\n   return mapIndexed { index, element ->\n   action(index, element)\n   element\n   }\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this sequence is empty. If the sequence can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun <S, T : S> Sequence<T>.reduce(operation: (acc: S, T) -> S): S {\n   val iterator = this.iterator()\n   if (!iterator.hasNext()) throw UnsupportedOperationException(\"Empty sequence can't be reduced.\")\n   var accumulator: S = iterator.next()\n   while (iterator.hasNext()) {\n   accumulator = operation(accumulator, iterator.next())\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original sequence.\n * \n * Throws an exception if this sequence is empty. If the sequence can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun <S, T : S> Sequence<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {\n   val iterator = this.iterator()\n   if (!iterator.hasNext()) throw UnsupportedOperationException(\"Empty sequence can't be reduced.\")\n   var index = 1\n   var accumulator: S = iterator.next()\n   while (iterator.hasNext()) {\n   accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original sequence.\n * \n * Returns `null` if the sequence is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <S, T : S> Sequence<T>.reduceIndexedOrNull(operation: (index: Int, acc: S, T) -> S): S? {\n   val iterator = this.iterator()\n   if (!iterator.hasNext()) return null\n   var index = 1\n   var accumulator: S = iterator.next()\n   while

(iterator.hasNext()) {\n accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the sequence is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <S, T : S> Sequence<T>.reduceOrNull(operation: (acc: S, T) -> S): S? {\n val iterator = this.iterator()\n if (!iterator.hasNext()) return null\n var accumulator: S = iterator.next()\n while (iterator.hasNext()) {\n accumulator = operation(accumulator, iterator.next())\n }\n return accumulator\n}\n\n/**\n * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting sequence.\n * The [initial] value should also be immutable (or should not be mutated)\n * as it may be passed to [operation] function later because of sequence's lazy nature.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\npublic fun <T, R> Sequence<T>.runningFold(initial: R, operation: (acc: R, T) -> R): Sequence<R> {\n return sequence {\n yield(initial)\n var accumulator = initial\n for (element in this@runningFold) {\n accumulator = operation(accumulator, element)\n yield(accumulator)\n }\n }\n}\n\n/**\n * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original sequence and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting sequence.\n * The [initial] value should also be immutable (or should not be mutated)\n * as it may be passed to [operation] function later because of sequence's lazy nature.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\npublic fun <T, R> Sequence<T>.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): Sequence<R> {\n return sequence {\n yield(initial)\n var index = 0\n var accumulator = initial\n for (element in this@runningFoldIndexed) {\n accumulator = operation(checkIndexOverflow(index++), accumulator, element)\n yield(accumulator)\n }\n }\n}\n\n/**\n * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this sequence.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting sequence.\n * \n * @param [operation] function that takes current accumulator value and the element, and calculates the next accumulator value.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun <S, T : S> Sequence<T>.runningReduce(operation: (acc: S, T) -> S): Sequence<S> {\n return sequence {\n val iterator = iterator()\n if (iterator.hasNext()) {\n var accumulator: S = iterator.next()\n yield(accumulator)\n while (iterator.hasNext()) {\n accumulator = operation(accumulator, iterator.next())\n yield(accumulator)\n }\n }\n }\n}\n\n/**\n * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting sequence.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n *\n * The operation is _intermediate_ and _stateless_.\n

* \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\npublic fun <S, T : S> Sequence<T>.runningReduceIndexed(operation: (index: Int, acc: S, T) -> S): Sequence<S> {\n return sequence {\n val iterator = iterator()\n if (iterator.hasNext()) {\n var accumulator: S = iterator.next()\n yield(accumulator)\n var index = 1\n while (iterator.hasNext()) {\n accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())\n yield(accumulator)\n }\n }\n }\n}\n\n/**\n * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting sequence.\n * The [initial] value should also be immutable (or should not be mutated)\n * as it may be passed to [operation] function later because of sequence's lazy nature.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun <T, R> Sequence<T>.scan(initial: R, operation: (acc: R, T) -> R): Sequence<R> {\n return runningFold(initial, operation)\n}\n\n/**\n * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original sequence and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting sequence.\n * The [initial] value should also be immutable (or should not be mutated)\n * as it may be passed to [operation] function later because of sequence's lazy nature.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun <T, R> Sequence<T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): Sequence<R> {\n return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the sequence.\n *\n * The operation is _terminal_.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun <T> Sequence<T>.sumBy(selector: (T) -> Int): Int {\n var sum: Int = 0\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the sequence.\n *\n * The operation is _terminal_.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun <T> Sequence<T>.sumByDouble(selector: (T) -> Double): Double {\n var sum: Double = 0.0\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the sequence.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.sumOf(selector: (T) -> Double): Double {\n var sum: Double = 0.toDouble()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the sequence.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.sumOf(selector: (T) -> Int): Int {\n var sum: Int = 0.toInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the sequence.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun

<T> Sequence<T>.sumOf(selector: (T) -> Long): Long {\n var sum: Long = 0.toLong()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the sequence.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.sumOf(selector: (T) -> UInt): UInt {\n var sum: UInt = 0.toUInt()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the sequence.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.sumOf(selector: (T) -> ULong): ULong {\n var sum: ULong = 0.toULong()\n for (element in this) {\n sum += selector(element)\n }\n return sum\n}\n\n/**\n * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic fun <T : Any> Sequence<T?>.requireNoNulls(): Sequence<T> {\n return map { it ?: throw IllegalArgumentException(\"null element found in $this.\") }\n}\n\n/**\n * Splits this sequence into a sequence of lists each not exceeding the given [size].\n * \n * The last list in the resulting sequence may have fewer elements than the given [size].\n * \n * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this sequence.\n *\n * The operation is _intermediate_ and _stateful_.\n * \n * @sample samples.collections.Collections.Transformations.chunked\n */\n@SinceKotlin(\"1.2\")\npublic fun <T> Sequence<T>.chunked(size: Int): Sequence<List<T>> {\n return windowed(size, size, partialWindows = true)\n}\n\n/**\n * Splits this sequence into several lists each not exceeding the given [size]\n * and applies the given [transform] function to an each.\n * \n * @return sequence of results of the [transform] applied to an each list.\n * \n * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * The last list may have fewer elements than the given [size].\n * \n * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this sequence.\n *\n * The operation is _intermediate_ and _stateful_.\n * \n * @sample samples.text.Strings.chunkedTransform\n */\n@SinceKotlin(\"1.2\")\npublic fun <T, R> Sequence<T>.chunked(size: Int, transform: (List<T>) -> R): Sequence<R> {\n return windowed(size, size, partialWindows = true, transform = transform)\n}\n\n/**\n * Returns a sequence containing all elements of the original sequence without the first occurrence of the given [element].\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic operator fun <T> Sequence<T>.minus(element: T): Sequence<T> {\n return object: Sequence<T> {\n override fun iterator(): Iterator<T> {\n var removed = false\n return this@minus.filter { if (!removed && it == element) { removed = true; false } else true }.iterator()\n }\n }\n}\n\n/**\n * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] array.\n * \n * Note that the source sequence and the array being subtracted are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n * \n * Before Kotlin 1.6, the [elements] array may have been converted to a [HashSet] to speed up the operation, thus the elements were required to have\n * a correct and stable implementation of `hashCode()` that didn't change between successive invocations.\n * On JVM, you can enable this behavior back with the system property `kotlin.collections.convert_arg_to_set_in_removeAll` set to `true`.\n *\n * The operation is _intermediate_ and _stateful_.\n */\npublic operator fun <T> Sequence<T>.minus(elements: Array<out T>): Sequence<T> {\n if (elements.isEmpty()) return this\n return object: Sequence<T> {\n override fun iterator(): Iterator<T> {\n val other = elements.convertToSetForSetOperation()\n return this@minus.filterNot { it in other }.iterator()\n }\n }\n}\n\n/**\n * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] collection.\n * \n * Note that the source sequence and the collection being

subtracted are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n * \n * Before Kotlin 1.6, the [elements] collection may have been converted to a [HashSet] to speed up the operation, thus the elements were required to have\n * a correct and stable implementation of `hashCode()` that didn't change between successive invocations.\n * On JVM, you can enable this behavior back with the system property `kotlin.collections.convert_arg_to_set_in_removeAll` set to `true`.\n *\n * The operation is _intermediate_ and _stateful_.\n */\npublic operator fun <T> Sequence<T>.minus(elements: Iterable<T>): Sequence<T> {\n return object: Sequence<T> {\n override fun iterator(): Iterator<T> {\n val other = elements.convertToSetForSetOperation()\n if (other.isEmpty())\n return this@minus.iterator()\n else\n return this@minus.filterNot { it in other }.iterator()\n }\n }\n}\n\n/**\n * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] sequence.\n * \n * Note that the source sequence and the sequence being subtracted are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n * \n * The operation is _intermediate_ for this sequence and _terminal_ and _stateful_ for the [elements] sequence.\n * \n * Before Kotlin 1.6, the [elements] sequence may have been converted to a [HashSet] to speed up the operation, thus the elements were required to have\n * a correct and stable implementation of `hashCode()` that didn't change between successive invocations.\n * On JVM, you can enable this behavior back with the system property `kotlin.collections.convert_arg_to_set_in_removeAll` set to `true`.\n */\npublic operator fun <T> Sequence<T>.minus(elements: Sequence<T>): Sequence<T> {\n return object: Sequence<T> {\n override fun iterator(): Iterator<T> {\n val other = elements.convertToSetForSetOperation()\n if (other.isEmpty())\n return this@minus.iterator()\n else\n return this@minus.filterNot { it in other }.iterator()\n }\n }\n}\n\n/**\n * Returns a sequence containing all elements of the original sequence without the first occurrence of the given [element].\n *\n * The operation is _intermediate_ and _stateless_.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.minusElement(element: T): Sequence<T> {\n return minus(element)\n}\n\n/**\n * Splits the original sequence into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains elements for which [predicate] yielded `false`.\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Sequences.Transformations.partition\n */\npublic inline fun <T> Sequence<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {\n val first = ArrayList<T>()\n val second = ArrayList<T>()\n for (element in this) {\n if (predicate(element)) {\n first.add(element)\n } else {\n second.add(element)\n }\n }\n return Pair(first, second)\n}\n\n/**\n * Returns a sequence containing all elements of the original sequence and then the given [element].\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic operator fun <T> Sequence<T>.plus(element: T): Sequence<T> {\n return sequenceOf(this, sequenceOf(element)).flatten()\n}\n\n/**\n * Returns a sequence containing all elements of original sequence and then all elements of the given [elements] array.\n * \n * Note that the source sequence and the array being added are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic operator fun <T> Sequence<T>.plus(elements: Array<out T>): Sequence<T> {\n return this.plus(elements.asList())\n}\n\n/**\n * Returns a sequence containing all elements of original sequence and then all elements of the given [elements] collection.\n * \n * Note that the source sequence and the collection being added are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic operator fun <T> Sequence<T>.plus(elements: Iterable<T>): Sequence<T> {\n return sequenceOf(this, elements.asSequence()).flatten()\n}\n\n/**\n * Returns a sequence containing all elements of original sequence and then all elements of the given [elements] sequence.\n * \n * Note that the source sequence and the sequence being added are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic operator fun <T> Sequence<T>.plus(elements: Sequence<T>): Sequence<T> {\n return sequenceOf(this, elements).flatten()\n}\n\n/**\n * Returns a sequence containing all elements of the original

sequence and then the given [element].\n *\n * The operation is _intermediate_ and _stateless_.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.plusElement(element: T): Sequence<T> {\n return plus(element)\n}\n\n/**\n * Returns a sequence of snapshots of the window of the given [size]\n * sliding along this sequence with the given [step], where each\n * snapshot is a list.\n * \n * Several last lists may have fewer elements than the given [size].\n * \n * Both [size] and [step] must be positive and can be greater than the number of elements in this sequence.\n * @param size the number of elements to take in each window\n * @param step the number of elements to move the window forward by on an each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample samples.collections.Sequences.Transformations.takeWindows\n */\n@SinceKotlin(\"1.2\")\npublic fun <T> Sequence<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): Sequence<List<T>> {\n return windowedSequence(size, step, partialWindows, reuseBuffer = false)\n}\n\n/**\n * Returns a sequence of results of applying the given [transform] function to\n * an each list representing a view over the window of the given [size]\n * sliding along this sequence with the given [step].\n * \n * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * Several last lists may have fewer elements than the given [size].\n * \n * Both [size] and [step] must be positive and can be greater than the number of elements in this sequence.\n * @param size the number of elements to take in each window\n * @param step the number of elements to move the window forward by on an each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample samples.collections.Sequences.Transformations.averageWindows\n */\n@SinceKotlin(\"1.2\")\npublic fun <T, R> Sequence<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (List<T>) -> R): Sequence<R> {\n return windowedSequence(size, step, partialWindows, reuseBuffer = true).map(transform)\n}\n\n/**\n * Returns a sequence of values built from the elements of `this` sequence and the [other] sequence with the same index.\n * The resulting sequence ends as soon as the shortest input sequence ends.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Sequences.Transformations.zip\n */\npublic infix fun <T, R> Sequence<T>.zip(other: Sequence<R>): Sequence<Pair<T, R>> {\n return MergingSequence(this, other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a sequence of values built from the elements of `this` sequence and the [other] sequence with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The resulting sequence ends as soon as the shortest input sequence ends.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Sequences.Transformations.zipWithTransform\n */\npublic fun <T, R, V> Sequence<T>.zip(other: Sequence<R>, transform: (a: T, b: R) -> V): Sequence<V> {\n return MergingSequence(this, other, transform)\n}\n\n/**\n * Returns a sequence of pairs of each two adjacent elements in this sequence.\n * \n * The returned sequence is empty if this sequence contains less than two elements.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Transformations.zipWithNext\n */\n@SinceKotlin(\"1.2\")\npublic fun <T> Sequence<T>.zipWithNext(): Sequence<Pair<T, T>> {\n return zipWithNext { a, b -> a to b }\n}\n\n/**\n * Returns a sequence containing the results of applying the given [transform] function\n * to an each pair of two adjacent elements in this sequence.\n * \n * The returned sequence is empty if this sequence contains less than two elements.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas\n */\n@SinceKotlin(\"1.2\")\npublic fun <T, R> Sequence<T>.zipWithNext(transform: (a: T, b: T) -> R): Sequence<R> {\n return sequence result@ {\n val iterator = iterator()\n if (!iterator.hasNext()) return@result\n var current = iterator.next()\n while (iterator.hasNext()) {\n val next = iterator.next()\n yield(transform(current, next))\n current = next\n }\n }\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the

[truncated] string (which defaults to \"...\").\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <T, A : Appendable> Sequence<T>.joinTo(buffer: A, separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((T) -> CharSequence)? = null): A {\n buffer.append(prefix)\n var count = 0\n for (element in this) {\n if (++count > 1) buffer.append(separator)\n if (limit < 0 || count <= limit) {\n buffer.appendElement(element, transform)\n } else break\n }\n if (limit >= 0 && count > limit) buffer.append(truncated)\n buffer.append(postfix)\n return buffer\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to \"...\").\n *\n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun <T> Sequence<T>.joinToString(separator: CharSequence = \", \", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = \"...\", transform: ((T) -> CharSequence)? = null): String {\n return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original sequence returning its elements when being iterated.\n */\npublic fun <T> Sequence<T>.asIterable(): Iterable<T> {\n return Iterable { this.iterator() }\n}\n\n/**\n * Returns this sequence as a [Sequence].\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.asSequence(): Sequence<T> {\n return this\n}\n\n/**\n * Returns an average value of elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"averageOfByte\")\npublic fun Sequence<Byte>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"averageOfShort\")\npublic fun Sequence<Short>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"averageOfInt\")\npublic fun Sequence<Int>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"averageOfLong\")\npublic fun Sequence<Long>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"averageOfFloat\")\npublic fun Sequence<Float>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"averageOfDouble\")\npublic fun Sequence<Double>.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n checkCountOverflow(++count)\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"sumOfByte\")\npublic fun Sequence<Byte>.sum(): Int {\n var sum: Int = 0\n for (element in this) {\n sum += element\n }\n return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"sumOfShort\")\npublic fun Sequence<Short>.sum(): Int {\n var sum: Int = 0\n for (element in this) {\n sum += element\n }\n return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"sumOfInt\")\npublic fun Sequence<Int>.sum(): Int {\n var sum: Int = 0\n for

(element in this) {\n sum += element\n }\n return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"sumOfLong\")\npublic fun Sequence<Long>.sum(): Long {\n var sum: Long = 0L\n for (element in this) {\n sum += element\n }\n return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"sumOfFloat\")\npublic fun Sequence<Float>.sum(): Float {\n var sum: Float = 0.0f\n for (element in this) {\n sum += element\n }\n return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"sumOfDouble\")\npublic fun Sequence<Double>.sum(): Double {\n var sum: Double = 0.0\n for (element in this) {\n sum += element\n }\n return sum\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"SetsKt\")\n\npackage kotlin.collections\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\nimport kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns a set containing all elements of the original set except the given [element].\n * \n * The returned set preserves the element iteration order of the original set.\n */\npublic operator fun <T> Set<T>.minus(element: T): Set<T> {\n val result = LinkedHashSet<T>(mapCapacity(size))\n var removed = false\n return this.filterTo(result) { if (!removed && it == element) { removed = true; false } else true }\n}\n\n/**\n * Returns a set containing all elements of the original set except the elements contained in the given [elements] array.\n * \n * The returned set preserves the element iteration order of the original set.\n * \n * Before Kotlin 1.6, the [elements] array may have been converted to a [HashSet] to speed up the operation, thus the elements were required to have\n * a correct and stable implementation of `hashCode()` that didn't change between successive invocations.\n * On JVM, you can enable this behavior back with the system property `kotlin.collections.convert_arg_to_set_in_removeAll` set to `true`.\n */\npublic operator fun <T> Set<T>.minus(elements: Array<out T>): Set<T> {\n val result = LinkedHashSet<T>(this)\n result.removeAll(elements)\n return result\n}\n\n/**\n * Returns a set containing all elements of the original set except the elements contained in the given [elements] collection.\n * \n * The returned set preserves the element iteration order of the original set.\n * \n * Before Kotlin 1.6, the [elements] collection may have been converted to a [HashSet] to speed up the operation, thus the elements were required to have\n * a correct and stable implementation of `hashCode()` that didn't change between successive invocations.\n * On JVM, you can enable this behavior back with the system property `kotlin.collections.convert_arg_to_set_in_removeAll` set to `true`.\n */\npublic operator fun <T> Set<T>.minus(elements: Iterable<T>): Set<T> {\n val other = elements.convertToSetForSetOperationWith(this)\n if (other.isEmpty())\n return this.toSet()\n if (other is Set)\n return this.filterNotTo(LinkedHashSet<T>()) { it in other }\n val result = LinkedHashSet<T>(this)\n result.removeAll(other)\n return result\n}\n\n/**\n * Returns a set containing all elements of the original set except the elements contained in the given [elements] sequence.\n * \n * The returned set preserves the element iteration order of the original set.\n * \n * Before Kotlin 1.6, the [elements] sequence may have been converted to a [HashSet] to speed up the operation, thus the elements were required to have\n * a correct and stable implementation of `hashCode()` that didn't change between successive invocations.\n * On JVM, you can enable this behavior back with the system property `kotlin.collections.convert_arg_to_set_in_removeAll` set to `true`.\n */\npublic operator fun <T> Set<T>.minus(elements: Sequence<T>): Set<T> {\n val result = LinkedHashSet<T>(this)\n result.removeAll(elements)\n return result\n}\n\n/**\n * Returns a set containing all elements of the original set except the given [element].\n * \n * The returned set preserves the element iteration order of the original set.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Set<T>.minusElement(element: T): Set<T> {\n return minus(element)\n}\n\n/**\n * Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.\n * \n * The returned set preserves the element iteration order of the original set.\n */\npublic operator fun <T> Set<T>.plus(element: T): Set<T> {\n val result = LinkedHashSet<T>(mapCapacity(size + 1))\n result.addAll(this)\n result.add(element)\n return

result\n}\n\n/**\n * Returns a set containing all elements of the original set and the given [elements] array,\n * which aren't already in this set.\n * \n * The returned set preserves the element iteration order of the original set.\n */\npublic operator fun <T> Set<T>.plus(elements: Array<out T>): Set<T> {\n   val result = LinkedHashSet<T>(mapCapacity(this.size + elements.size))\n   result.addAll(this)\n   result.addAll(elements)\n   return result\n}\n\n/**\n * Returns a set containing all elements of the original set and the given [elements] collection,\n * which aren't already in this set.\n * The returned set preserves the element iteration order of the original set.\n */\npublic operator fun <T> Set<T>.plus(elements: Iterable<T>): Set<T> {\n   val result = LinkedHashSet<T>(mapCapacity(elements.collectionSizeOrNull()?.let { this.size + it } ?: this.size * 2))\n   result.addAll(this)\n   result.addAll(elements)\n   return result\n}\n\n/**\n * Returns a set containing all elements of the original set and the given [elements] sequence,\n * which aren't already in this set.\n * \n * The returned set preserves the element iteration order of the original set.\n */\npublic operator fun <T> Set<T>.plus(elements: Sequence<T>): Set<T> {\n   val result = LinkedHashSet<T>(mapCapacity(this.size * 2))\n   result.addAll(this)\n   result.addAll(elements)\n   return result\n}\n\n/**\n * Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.\n * \n * The returned set preserves the element iteration order of the original set.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Set<T>.plusElement(element: T): Set<T> {\n   return plus(element)\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"StringsKt\")\n\npackage kotlin.text\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\n\n/**\n * Returns a character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char sequence.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic expect fun CharSequence.elementAt(index: Int): Char\n\n/**\n * Returns a character at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this char sequence.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.elementAtOrElse(index: Int, defaultValue: (Int) -> Char): Char {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.elementAtOrNull(index: Int): Char? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns the first character matching the given [predicate], or `null` if no such character was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.find(predicate: (Char) -> Boolean): Char? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the last character matching the given [predicate], or `null` if no such character was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.findLast(predicate: (Char) -> Boolean): Char? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns first character.\n * @throws [NoSuchElementException] if the char sequence is empty.\n */\npublic fun CharSequence.first(): Char {\n   if (isEmpty())\n   throw NoSuchElementException(\"Char sequence is empty.\")\n   return this[0]\n}\n\n/**\n * Returns the first character matching the given [predicate].\n * @throws [NoSuchElementException] if no such character is found.\n */\npublic inline fun CharSequence.first(predicate: (Char) -> Boolean): Char {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Char sequence contains no character matching the predicate.\")\n}\n\n/**\n * Returns the first non-null value produced by [transform] function being applied to characters of this char sequence in iteration order,\n * or throws [NoSuchElementException] if no non-null value was produced.\n * \n * @sample samples.collections.Collections.Transformations.firstNotNullOf\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun <R : Any> CharSequence.firstNotNullOf(transform: (Char) -> R?): R {\n   return firstNotNullOfOrNull(transform) ?: throw

NoSuchElementException(\"No element of the char sequence was transformed to a non-null value.\")\n}\n\n/**\n * Returns the first non-null value produced by [transform] function being applied to characters of this char sequence in iteration order,\n * or `null` if no non-null value was produced.\n * \n * @sample samples.collections.Collections.Transformations.firstNotNullOf\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun <R : Any> CharSequence.firstNotNullOfOrNull(transform: (Char) -> R?): R? {\n for (element in this) {\n val result = transform(element)\n if (result != null) {\n return result\n }\n }\n return null\n}\n\n/**\n * Returns the first character, or `null` if the char sequence is empty.\n */\npublic fun CharSequence.firstOrNull(): Char? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first character matching the given [predicate], or `null` if character was not found.\n */\npublic inline fun CharSequence.firstOrNull(predicate: (Char) -> Boolean): Char? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns a character at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this char sequence.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {\n return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n */\npublic fun CharSequence.getOrNull(index: Int): Char? {\n return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns index of the first character matching the given [predicate], or -1 if the char sequence does not contain such character.\n */\npublic inline fun CharSequence.indexOfFirst(predicate: (Char) -> Boolean): Int {\n for (index in indices) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns index of the last character matching the given [predicate], or -1 if the char sequence does not contain such character.\n */\npublic inline fun CharSequence.indexOfLast(predicate: (Char) -> Boolean): Int {\n for (index in indices.reversed()) {\n if (predicate(this[index])) {\n return index\n }\n }\n return -1\n}\n\n/**\n * Returns the last character.\n * \n * @throws NoSuchElementException if the char sequence is empty.\n * \n * @sample samples.text.Strings.last\n */\npublic fun CharSequence.last(): Char {\n if (isEmpty())\n throw NoSuchElementException(\"Char sequence is empty.\")\n return this[lastIndex]\n}\n\n/**\n * Returns the last character matching the given [predicate].\n * \n * @throws NoSuchElementException if no such character is found.\n * \n * @sample samples.text.Strings.last\n */\npublic inline fun CharSequence.last(predicate: (Char) -> Boolean): Char {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n throw NoSuchElementException(\"Char sequence contains no character matching the predicate.\")\n}\n\n/**\n * Returns the last character, or `null` if the char sequence is empty.\n * \n * @sample samples.text.Strings.last\n */\npublic fun CharSequence.lastOrNull(): Char? {\n return if (isEmpty()) null else this[length - 1]\n}\n\n/**\n * Returns the last character matching the given [predicate], or `null` if no such character was found.\n * \n * @sample samples.text.Strings.last\n */\npublic inline fun CharSequence.lastOrNull(predicate: (Char) -> Boolean): Char? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns a random character from this char sequence.\n * \n * @throws NoSuchElementException if this char sequence is empty.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.random(): Char {\n return random(Random)\n}\n\n/**\n * Returns a random character from this char sequence using the specified source of randomness.\n * \n * @throws NoSuchElementException if this char sequence is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun CharSequence.random(random: Random): Char {\n if (isEmpty())\n throw NoSuchElementException(\"Char sequence is empty.\")\n return get(random.nextInt(length))\n}\n\n/**\n * Returns a random character from this char sequence, or `null` if this char sequence is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.randomOrNull(): Char? {\n return randomOrNull(Random)\n}\n\n/**\n * Returns a random character from this char sequence using the specified source of randomness, or `null` if this char sequence is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun CharSequence.randomOrNull(random: Random): Char? {\n if (isEmpty())\n return null\n return

get(random.nextInt(length))\n}\n\n/**\n * Returns the single character, or throws an exception if the char sequence is empty or has more than one character.\n */\npublic fun CharSequence.single(): Char {\n return when (length) {\n 0 -> throw NoSuchElementException(\"Char sequence is empty.\")\n 1 -> this[0]\n else -> throw IllegalArgumentException(\"Char sequence has more than one element.\")\n }\n}\n\n/**\n * Returns the single character matching the given [predicate], or throws exception if there is no or more than one matching character.\n */\npublic inline fun CharSequence.single(predicate: (Char) -> Boolean): Char {\n var single: Char? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Char sequence contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Char sequence contains no character matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as Char\n}\n\n/**\n * Returns single character, or `null` if the char sequence is empty or has more than one character.\n */\npublic fun CharSequence.singleOrNull(): Char? {\n return if (length == 1) this[0] else null\n}\n\n/**\n * Returns the single character matching the given [predicate], or `null` if character was not found or more than one character was found.\n */\npublic inline fun CharSequence.singleOrNull(predicate: (Char) -> Boolean): Char? {\n var single: Char? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) return null\n single = element\n found = true\n }\n }\n if (!found) return null\n return single\n}\n\n/**\n * Returns a subsequence of this char sequence with the first [n] characters removed.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.drop\n */\npublic fun CharSequence.drop(n: Int): CharSequence {\n require(n >= 0) { \"Requested character count $n is less than zero.\" }\n return subSequence(n.coerceAtMost(length), length)\n}\n\n/**\n * Returns a string with the first [n] characters removed.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.drop\n */\npublic fun String.drop(n: Int): String {\n require(n >= 0) { \"Requested character count $n is less than zero.\" }\n return substring(n.coerceAtMost(length))\n}\n\n/**\n * Returns a subsequence of this char sequence with the last [n] characters removed.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.drop\n */\npublic fun CharSequence.dropLast(n: Int): CharSequence {\n require(n >= 0) { \"Requested character count $n is less than zero.\" }\n return take((length - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a string with the last [n] characters removed.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.drop\n */\npublic fun String.dropLast(n: Int): String {\n require(n >= 0) { \"Requested character count $n is less than zero.\" }\n return take((length - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a subsequence of this char sequence containing all characters except last characters that satisfy the given [predicate].\n * \n * @sample samples.text.Strings.drop\n */\npublic inline fun CharSequence.dropLastWhile(predicate: (Char) -> Boolean): CharSequence {\n for (index in lastIndex downTo 0)\n if (!predicate(this[index]))\n return subSequence(0, index + 1)\n return \"\"\n}\n\n/**\n * Returns a string containing all characters except last characters that satisfy the given [predicate].\n * \n * @sample samples.text.Strings.drop\n */\npublic inline fun String.dropLastWhile(predicate: (Char) -> Boolean): String {\n for (index in lastIndex downTo 0)\n if (!predicate(this[index]))\n return substring(0, index + 1)\n return \"\"\n}\n\n/**\n * Returns a subsequence of this char sequence containing all characters except first characters that satisfy the given [predicate].\n * \n * @sample samples.text.Strings.drop\n */\npublic inline fun CharSequence.dropWhile(predicate: (Char) -> Boolean): CharSequence {\n for (index in this.indices)\n if (!predicate(this[index]))\n return subSequence(index, length)\n return \"\"\n}\n\n/**\n * Returns a string containing all characters except first characters that satisfy the given [predicate].\n * \n * @sample samples.text.Strings.drop\n */\npublic inline fun String.dropWhile(predicate: (Char) -> Boolean): String {\n for (index in this.indices)\n if (!predicate(this[index]))\n return substring(index)\n return \"\"\n}\n\n/**\n * Returns a char sequence containing only those characters from the original char sequence that match the given [predicate].\n * \n * @sample samples.text.Strings.filter\n */\npublic inline fun CharSequence.filter(predicate: (Char) -> Boolean): CharSequence {\n return filterTo(StringBuilder(), predicate)\n}\n\n/**\n * Returns a string containing only those characters from the original string that match the given [predicate].\n * \n * @sample samples.text.Strings.filter\n */\npublic inline fun String.filter(predicate: (Char) -

> Boolean): String {\n return filterTo(StringBuilder(), predicate).toString()\n}\n\n/**\n * Returns a char sequence containing only those characters from the original char sequence that match the given [predicate].\n * @param [predicate] function that takes the index of a character and the character itself\n * and returns the result of predicate evaluation on the character.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun CharSequence.filterIndexed(predicate: (index: Int, Char) -> Boolean): CharSequence {\n return filterIndexedTo(StringBuilder(), predicate)\n}\n\n/**\n * Returns a string containing only those characters from the original string that match the given [predicate].\n * @param [predicate] function that takes the index of a character and the character itself\n * and returns the result of predicate evaluation on the character.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n */\npublic inline fun String.filterIndexed(predicate: (index: Int, Char) -> Boolean): String {\n return filterIndexedTo(StringBuilder(), predicate).toString()\n}\n\n/**\n * Appends all characters matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of a character and the character itself\n * and returns the result of predicate evaluation on the character.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\npublic inline fun <C : Appendable> CharSequence.filterIndexedTo(destination: C, predicate: (index: Int, Char) -> Boolean): C {\n forEachIndexed { index, element ->\n if (predicate(index, element)) destination.append(element)\n }\n return destination\n}\n\n/**\n * Returns a char sequence containing only those characters from the original char sequence that do not match the given [predicate].\n * \n * @sample samples.text.Strings.filterNot\n */\npublic inline fun CharSequence.filterNot(predicate: (Char) -> Boolean): CharSequence {\n return filterNotTo(StringBuilder(), predicate)\n}\n\n/**\n * Returns a string containing only those characters from the original string that do not match the given [predicate].\n * \n * @sample samples.text.Strings.filterNot\n */\npublic inline fun String.filterNot(predicate: (Char) -> Boolean): String {\n return filterNotTo(StringBuilder(), predicate).toString()\n}\n\n/**\n * Appends all characters not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : Appendable> CharSequence.filterNotTo(destination: C, predicate: (Char) -> Boolean): C {\n for (element in this) if (!predicate(element)) destination.append(element)\n return destination\n}\n\n/**\n * Appends all characters matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : Appendable> CharSequence.filterTo(destination: C, predicate: (Char) -> Boolean): C {\n for (index in 0 until length) {\n val element = get(index)\n if (predicate(element)) destination.append(element)\n }\n return destination\n}\n\n/**\n * Returns a char sequence containing characters of the original char sequence at the specified range of [indices].\n */\npublic fun CharSequence.slice(indices: IntRange): CharSequence {\n if (indices.isEmpty()) return \"\"\n return subSequence(indices)\n}\n\n/**\n * Returns a string containing characters of the original string at the specified range of [indices].\n */\npublic fun String.slice(indices: IntRange): String {\n if (indices.isEmpty()) return \"\"\n return substring(indices)\n}\n\n/**\n * Returns a char sequence containing characters of the original char sequence at specified [indices].\n */\npublic fun CharSequence.slice(indices: Iterable<Int>): CharSequence {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return \"\"\n val result = StringBuilder(size)\n for (i in indices) {\n result.append(get(i))\n }\n return result\n}\n\n/**\n * Returns a string containing characters of the original string at specified [indices].\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.slice(indices: Iterable<Int>): String {\n return (this as CharSequence).slice(indices).toString()\n}\n\n/**\n * Returns a subsequence of this char sequence containing the first [n] characters from this char sequence, or the entire char sequence if this char sequence is shorter.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.take\n */\npublic fun CharSequence.take(n: Int): CharSequence {\n require(n >= 0) { \"Requested character count $n is less than zero.\" }\n return subSequence(0, n.coerceAtMost(length))\n}\n\n/**\n * Returns a string containing the first [n] characters from this string, or the entire string if this string is shorter.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.take\n */\npublic fun String.take(n: Int): String {\n require(n >= 0) { \"Requested character count $n is less than zero.\" }\n return substring(0, n.coerceAtMost(length))\n}\n\n/**\n * Returns a subsequence of this char sequence containing the last [n]

characters from this char sequence, or the entire char sequence if this char sequence is shorter.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.take\n */\npublic fun CharSequence.takeLast(n: Int): CharSequence {\n   require(n >= 0) { \"Requested character count $n is less than zero.\" }\n   val length = length\n   return subSequence(length - n.coerceAtMost(length), length)\n}\n\n/**\n * Returns a string containing the last [n] characters from this string, or the entire string if this string is shorter.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.take\n */\npublic fun String.takeLast(n: Int): String {\n   require(n >= 0) { \"Requested character count $n is less than zero.\" }\n   val length = length\n   return substring(length - n.coerceAtMost(length))\n}\n\n/**\n * Returns a subsequence of this char sequence containing last characters that satisfy the given [predicate].\n * \n * @sample samples.text.Strings.take\n */\npublic inline fun CharSequence.takeLastWhile(predicate: (Char) -> Boolean): CharSequence {\n   for (index in lastIndex downTo 0) {\n   if (!predicate(this[index])) {\n   return subSequence(index + 1, length)\n   }\n   }\n   return subSequence(0, length)\n}\n\n/**\n * Returns a string containing last characters that satisfy the given [predicate].\n * \n * @sample samples.text.Strings.take\n */\npublic inline fun String.takeLastWhile(predicate: (Char) -> Boolean): String {\n   for (index in lastIndex downTo 0) {\n   if (!predicate(this[index])) {\n   return substring(index + 1)\n   }\n   }\n   return this\n}\n\n/**\n * Returns a subsequence of this char sequence containing the first characters that satisfy the given [predicate].\n * \n * @sample samples.text.Strings.take\n */\npublic inline fun CharSequence.takeWhile(predicate: (Char) -> Boolean): CharSequence {\n   for (index in 0 until length)\n   if (!predicate(get(index))) {\n   return subSequence(0, index)\n   }\n   return subSequence(0, length)\n}\n\n/**\n * Returns a string containing the first characters that satisfy the given [predicate].\n * \n * @sample samples.text.Strings.take\n */\npublic inline fun String.takeWhile(predicate: (Char) -> Boolean): String {\n   for (index in 0 until length)\n   if (!predicate(get(index))) {\n   return substring(0, index)\n   }\n   return this\n}\n\n/**\n * Returns a char sequence with characters in reversed order.\n */\npublic fun CharSequence.reversed(): CharSequence {\n   return StringBuilder(this).reverse()\n}\n\n/**\n * Returns a string with characters in reversed order.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.reversed(): String {\n   return (this as CharSequence).reversed().toString()\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to characters of the given char sequence.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original char sequence.\n * \n * @sample samples.text.Strings.associate\n */\npublic inline fun <K, V> CharSequence.associate(transform: (Char) -> Pair<K, V>): Map<K, V> {\n   val capacity = mapCapacity(length).coerceAtLeast(16)\n   return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing the characters from the given char sequence indexed by the key\n * returned from [keySelector] function applied to each character.\n * \n * If any two characters would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original char sequence.\n * \n * @sample samples.text.Strings.associateBy\n */\npublic inline fun <K> CharSequence.associateBy(keySelector: (Char) -> K): Map<K, Char> {\n   val capacity = mapCapacity(length).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, Char>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to characters of the given char sequence.\n * \n * If any two characters would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original char sequence.\n * \n * @sample samples.text.Strings.associateByWithValueTransform\n */\npublic inline fun <K, V> CharSequence.associateBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {\n   val capacity = mapCapacity(length).coerceAtLeast(16)\n   return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each character of the given char sequence\n * and value is the character itself.\n * \n * If any two characters would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.text.Strings.associateByTo\n */\npublic inline fun <K,

M : MutableMap<in K, in Char>> CharSequence.associateByTo(destination: M, keySelector: (Char) -> K): M {\n for (element in this) {\n destination.put(keySelector(element), element)\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to characters of the given char sequence.\n * \n * If any two characters would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * @sample samples.text.Strings.associateByToWithValueTransform\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> CharSequence.associateByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {\n for (element in this) {\n destination.put(keySelector(element), valueTransform(element))\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each character of the given char sequence.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * @sample samples.text.Strings.associateTo\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> CharSequence.associateTo(destination: M, transform: (Char) -> Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n return destination\n}\n\n/**\n * Returns a [Map] where keys are characters from the given char sequence and values are\n * produced by the [valueSelector] function applied to each character.\n * \n * If any two characters are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original char sequence.\n * \n * @sample samples.text.Strings.associateWith\n */\n@SinceKotlin(\"1.3\")\npublic inline fun <V> CharSequence.associateWith(valueSelector: (Char) -> V): Map<Char, V> {\n val result = LinkedHashMap<Char, V>(mapCapacity(length.coerceAtMost(128)).coerceAtLeast(16))\n return associateWithTo(result, valueSelector)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each character of the given char sequence,\n * where key is the character itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two characters are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.text.Strings.associateWithTo\n */\n@SinceKotlin(\"1.3\")\npublic inline fun <V, M : MutableMap<in Char, in V>> CharSequence.associateWithTo(destination: M, valueSelector: (Char) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Appends all characters to the given [destination] collection.\n */\npublic fun <C : MutableCollection<in Char>> CharSequence.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Returns a new [HashSet] of all characters.\n */\npublic fun CharSequence.toHashSet(): HashSet<Char> {\n return toCollection(HashSet<Char>(mapCapacity(length.coerceAtMost(128))))\n}\n\n/**\n * Returns a [List] containing all characters.\n */\npublic fun CharSequence.toList(): List<Char> {\n return when (length) {\n 0 -> emptyList()\n 1 -> listOf(this[0])\n else -> this.toMutableList()\n }\n}\n\n/**\n * Returns a new [MutableList] filled with all characters of this char sequence.\n */\npublic fun CharSequence.toMutableList(): MutableList<Char> {\n return toCollection(ArrayList<Char>(length))\n}\n\n/**\n * Returns a [Set] of all characters.\n * \n * The returned set preserves the element iteration order of the original char sequence.\n */\npublic fun CharSequence.toSet(): Set<Char> {\n return when (length) {\n 0 -> emptySet()\n 1 -> setOf(this[0])\n else -> toCollection(LinkedHashSet<Char>(mapCapacity(length.coerceAtMost(128))))\n }\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each character of original char sequence.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\npublic inline fun <R> CharSequence.flatMap(transform: (Char) -> Iterable<R>): List<R> {\n return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each character\n * and its index in the original char sequence.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterable\")\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharSequence.flatMapIndexed(transform: (index: Int, Char) -> Iterable<R>): List<R> {\n return

flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each character\n * and its index in the original char sequence, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"flatMapIndexedIterableTo\")\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> CharSequence.flatMapIndexedTo(destination: C, transform: (index: Int, Char) -> Iterable<R>): C {\n   var index = 0\n   for (element in this) {\n   val list = transform(index++, element)\n   destination.addAll(list)\n   }\n   return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each character of original char sequence, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> CharSequence.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {\n   for (element in this) {\n   val list = transform(element)\n   destination.addAll(list)\n   }\n   return destination\n}\n\n/**\n * Groups characters of the original char sequence by the key returned by the given [keySelector] function\n * applied to each character and returns a map where each group key is associated with a list of corresponding characters.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original char sequence.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K> CharSequence.groupBy(keySelector: (Char) -> K): Map<K, List<Char>> {\n   return groupByTo(LinkedHashMap<K, MutableList<Char>>(), keySelector)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each character of the original char sequence\n * by the key returned by the given [keySelector] function applied to the character\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original char sequence.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V> CharSequence.groupBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>> {\n   return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups characters of the original char sequence by the key returned by the given [keySelector] function\n * applied to each character and puts to the [destination] map each group key associated with a list of corresponding characters.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K, M : MutableMap<in K, MutableList<Char>>> CharSequence.groupByTo(destination: M, keySelector: (Char) -> K): M {\n   for (element in this) {\n   val key = keySelector(element)\n   val list = destination.getOrPut(key) { ArrayList<Char>() }\n   list.add(element)\n   }\n   return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each character of the original char sequence\n * by the key returned by the given [keySelector] function applied to the character\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> CharSequence.groupByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {\n   for (element in this) {\n   val key = keySelector(element)\n   val list = destination.getOrPut(key) { ArrayList<V>() }\n   list.add(valueTransform(element))\n   }\n   return destination\n}\n\n/**\n * Creates a [Grouping] source from a char sequence to be used later with one of group-and-fold operations\n * using the specified [keySelector] function to extract a key from each character.\n * \n * @sample samples.collections.Grouping.groupingByEachCount\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <K> CharSequence.groupingBy(crossinline keySelector: (Char) -> K): Grouping<Char, K> {\n   return object : Grouping<Char, K> {\n   override fun sourceIterator(): Iterator<Char> = this@groupingBy.iterator()\n   override fun keyOf(element: Char): K = keySelector(element)\n   }\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each character in the original char sequence.\n * \n * @sample samples.text.Strings.map\n */\npublic inline fun <R> CharSequence.map(transform: (Char) -> R): List<R> {\n   return mapTo(ArrayList<R>(length), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each character and its index in the original char sequence.\n *

@param [transform] function that takes the index of a character and the character itself\n * and returns the result of the transform applied to the character.\n */\npublic inline fun <R> CharSequence.mapIndexed(transform: (index: Int, Char) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(length), transform)\n}\n\n/**\n * Returns a list containing only the non-null results of applying the given [transform] function\n * to each character and its index in the original char sequence.\n * @param [transform] function that takes the index of a character and the character itself\n * and returns the result of the transform applied to the character.\n */\npublic inline fun <R : Any> CharSequence.mapIndexedNotNull(transform: (index: Int, Char) -> R?): List<R> {\n   return mapIndexedNotNullTo(ArrayList<R>(), transform)\n}\n\n/**\n * Applies the given [transform] function to each character and its index in the original char sequence\n * and appends only the non-null results to the given [destination].\n * @param [transform] function that takes the index of a character and the character itself\n * and returns the result of the transform applied to the character.\n */\npublic inline fun <R : Any, C : MutableCollection<in R>> CharSequence.mapIndexedNotNullTo(destination: C, transform: (index: Int, Char) -> R?): C {\n   forEachIndexed { index, element -> transform(index, element)?.let { destination.add(it) } }\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each character and its index in the original char sequence\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of a character and the character itself\n * and returns the result of the transform applied to the character.\n */\npublic inline fun <R, C : MutableCollection<in R>> CharSequence.mapIndexedTo(destination: C, transform: (index: Int, Char) -> R): C {\n   var index = 0\n   for (item in this)\n   destination.add(transform(index++, item))\n   return destination\n}\n\n/**\n * Returns a list containing only the non-null results of applying the given [transform] function\n * to each character in the original char sequence.\n * \n * @sample samples.collections.Collections.Transformations.mapNotNull\n */\npublic inline fun <R : Any> CharSequence.mapNotNull(transform: (Char) -> R?): List<R> {\n   return mapNotNullTo(ArrayList<R>(), transform)\n}\n\n/**\n * Applies the given [transform] function to each character in the original char sequence\n * and appends only the non-null results to the given [destination].\n */\npublic inline fun <R : Any, C : MutableCollection<in R>> CharSequence.mapNotNullTo(destination: C, transform: (Char) -> R?): C {\n   forEach { element -> transform(element)?.let { destination.add(it) } }\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each character of the original char sequence\n * and appends the results to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> CharSequence.mapTo(destination: C, transform: (Char) -> R): C {\n   for (item in this)\n   destination.add(transform(item))\n   return destination\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each character of the original char sequence\n * into an [IndexedValue] containing the index of that character and the character itself.\n */\npublic fun CharSequence.withIndex(): Iterable<IndexedValue<Char>> {\n   return IndexingIterable { iterator() }\n}\n\n/**\n * Returns `true` if all characters match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun CharSequence.all(predicate: (Char) -> Boolean): Boolean {\n   for (element in this) if (!predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if char sequence has at least one character.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun CharSequence.any(): Boolean {\n   return !isEmpty()\n}\n\n/**\n * Returns `true` if at least one character matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun CharSequence.any(predicate: (Char) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return true\n   return false\n}\n\n/**\n * Returns the length of this char sequence.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.count(): Int {\n   return length\n}\n\n/**\n * Returns the number of characters matching the given [predicate].\n */\npublic inline fun CharSequence.count(predicate: (Char) -> Boolean): Int {\n   var count = 0\n   for (element in this) if (predicate(element)) ++count\n   return count\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each character.\n * \n * Returns the specified [initial] value if the char sequence is empty.\n * \n * @param [operation] function that takes current accumulator value and a character, and calculates the next accumulator value.\n */\npublic inline fun <R> CharSequence.fold(initial: R, operation: (acc: R, Char) -> R): R {\n   var accumulator = initial\n   for (element in

this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each character with its index in the original char sequence.\n * \n * Returns the specified [initial] value if the char sequence is empty.\n * \n * @param [operation] function that takes the index of a character, current accumulator value\n * and the character itself, and calculates the next accumulator value.\n */\npublic inline fun <R> CharSequence.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {\n   var index = 0\n   var accumulator = initial\n   for (element in this) accumulator = operation(index++, accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each character and current accumulator value.\n * \n * Returns the specified [initial] value if the char sequence is empty.\n * \n * @param [operation] function that takes a character and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> CharSequence.foldRight(initial: R, operation: (Char, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each character with its index in the original char sequence and current accumulator value.\n * \n * Returns the specified [initial] value if the char sequence is empty.\n * \n * @param [operation] function that takes the index of a character, the character itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> CharSequence.foldRightIndexed(initial: R, operation: (index: Int, Char, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Performs the given [action] on each character.\n */\npublic inline fun CharSequence.forEach(action: (Char) -> Unit): Unit {\n   for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each character, providing sequential index with the character.\n * @param [action] function that takes the index of a character and the character itself\n * and performs the action on the character.\n */\npublic inline fun CharSequence.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {\n   var index = 0\n   for (item in this) action(index++, item)\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun CharSequence.max(): Char? {\n   return maxOrNull()\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> CharSequence.maxBy(selector: (Char) -> R): Char? {\n   return maxByOrNull(selector)\n}\n\n/**\n * Returns the first character yielding the largest value of the given function or `null` if there are no characters.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> CharSequence.maxByOrNull(selector: (Char) -> R): Char? {\n   if (isEmpty()) return null\n   var maxElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return maxElem\n   var maxValue = selector(maxElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (maxValue < v) {\n   maxElem = e\n   maxValue = v\n   }\n   }\n   return maxElem\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each character in the char sequence.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the char sequence is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.maxOf(selector: (Char) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each character in the char sequence.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the char sequence is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.maxOf(selector: (Char) -> Float): Float {\n if (isEmpty()) throw NoSuchElementException()\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each character in the char sequence.\n * \n * @throws NoSuchElementException if the char sequence is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> CharSequence.maxOf(selector: (Char) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (maxValue < v) {\n maxValue = v\n }\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each character in the char sequence or `null` if there are no characters.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.maxOfOrNull(selector: (Char) -> Double): Double? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each character in the char sequence or `null` if there are no characters.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.maxOfOrNull(selector: (Char) -> Float): Float? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each character in the char sequence or `null` if there are no characters.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> CharSequence.maxOfOrNull(selector: (Char) -> R): R? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (maxValue < v) {\n maxValue = v\n }\n }\n return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each character in the char sequence.\n * \n * @throws NoSuchElementException if the char sequence is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharSequence.maxOfWith(comparator: Comparator<in R>, selector: (Char) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(maxValue, v) < 0) {\n maxValue = v\n }\n }\n return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each character in the char sequence or `null` if there are no characters.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharSequence.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(maxValue, v) < 0) {\n maxValue = v\n }\n }\n return maxValue\n}\n\n/**\n * Returns the largest character or `null` if there are no characters.\n */\n@SinceKotlin(\"1.4\")\npublic fun CharSequence.maxOrNull(): Char? {\n if (isEmpty()) return null\n var max = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (max < e) max = e\n }\n return max\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\",

errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun CharSequence.maxWith(comparator: Comparator<in Char>): Char? {\n   return maxWithOrNull(comparator)\n}\n\n/**\n * Returns the first character having the largest value according to the provided [comparator] or `null` if there are no characters.\n */\n@SinceKotlin(\"1.4\")\npublic fun CharSequence.maxWithOrNull(comparator: Comparator<in Char>): Char? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(max, e) < 0) max = e\n   }\n   return max\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun CharSequence.min(): Char? {\n   return minOrNull()\n}\n\n@Deprecated(\"Use minByOrNull instead.\", ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic inline fun <R : Comparable<R>> CharSequence.minBy(selector: (Char) -> R): Char? {\n   return minByOrNull(selector)\n}\n\n/**\n * Returns the first character yielding the smallest value of the given function or `null` if there are no characters.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R : Comparable<R>> CharSequence.minByOrNull(selector: (Char) -> R): Char? {\n   if (isEmpty()) return null\n   var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue = selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v) {\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each character in the char sequence.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the char sequence is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.minOf(selector: (Char) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each character in the char sequence.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the char sequence is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.minOf(selector: (Char) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each character in the char sequence.\n * \n * @throws NoSuchElementException if the char sequence is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> CharSequence.minOf(selector: (Char) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each character in the char sequence or `null` if there are no characters.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.minOfOrNull(selector: (Char) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each character in the char sequence or `null` if there are no characters.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.minOfOrNull(selector: (Char) -> Float): Float? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each character in the char sequence or `null` if there are no characters.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> CharSequence.minOfOrNull(selector: (Char) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (minValue > v) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each character in the char sequence.\n * \n * @throws NoSuchElementException if the char sequence is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharSequence.minOfWith(comparator: Comparator<in R>, selector: (Char) -> R): R {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each character in the char sequence or `null` if there are no characters.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R> CharSequence.minOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {\n if (isEmpty()) return null\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n if (comparator.compare(minValue, v) > 0) {\n minValue = v\n }\n }\n return minValue\n}\n\n/**\n * Returns the smallest character or `null` if there are no characters.\n */\n@SinceKotlin(\"1.4\")\npublic fun CharSequence.minOrNull(): Char? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (min > e) min = e\n }\n return min\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\npublic fun CharSequence.minWith(comparator: Comparator<in Char>): Char? {\n return minWithOrNull(comparator)\n}\n\n/**\n * Returns the first character having the smallest value according to the provided [comparator] or `null` if there are no characters.\n */\n@SinceKotlin(\"1.4\")\npublic fun CharSequence.minWithOrNull(comparator: Comparator<in Char>): Char? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns `true` if the char sequence has no characters.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\npublic fun CharSequence.none(): Boolean {\n return isEmpty()\n}\n\n/**\n * Returns `true` if no characters match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun CharSequence.none(predicate: (Char) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return false\n return true\n}\n\n/**\n * Performs the given [action] on each character and returns the char sequence itself afterwards.\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <S : CharSequence> S.onEach(action: (Char) -> Unit): S {\n return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each character, providing sequential index with the character,\n * and returns the char sequence itself afterwards.\n * @param [action] function that takes the index of a character and the character itself\n * and performs the action on the character.\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <S : CharSequence> S.onEachIndexed(action: (index: Int, Char) -> Unit): S {\n return apply { forEachIndexed(action) }\n}\n\n/**\n * Accumulates value starting with the first character and applying [operation] from left to right\n * to current accumulator value and each character.\n * \n * Throws an exception if this char sequence is empty. If the char sequence can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function

that takes current accumulator value and a character,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun CharSequence.reduce(operation: (acc: Char, Char) -> Char): Char {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty char sequence can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first character and applying [operation] from left to right\n * to current accumulator value and each character with its index in the original char sequence.\n * \n * Throws an exception if this char sequence is empty. If the char sequence can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of a character, current accumulator value and the character itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\npublic inline fun CharSequence.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty char sequence can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first character and applying [operation] from left to right\n * to current accumulator value and each character with its index in the original char sequence.\n * \n * Returns `null` if the char sequence is empty.\n * \n * @param [operation] function that takes the index of a character, current accumulator value and the character itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun CharSequence.reduceIndexedOrNull(operation: (index: Int, acc: Char, Char) -> Char): Char? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first character and applying [operation] from left to right\n * to current accumulator value and each character.\n * \n * Returns `null` if the char sequence is empty.\n * \n * @param [operation] function that takes current accumulator value and a character,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun CharSequence.reduceOrNull(operation: (acc: Char, Char) -> Char): Char? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last character and applying [operation] from right to left\n * to each character and current accumulator value.\n * \n * Throws an exception if this char sequence is empty. If the char sequence can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes a character and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun CharSequence.reduceRight(operation: (Char, acc: Char) -> Char): Char {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty char sequence can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last character and applying [operation] from right to left\n * to each character with its index in the original char sequence and current accumulator value.\n * \n * Throws an exception if this char sequence is empty. If the char sequence can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of a character, the character itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\npublic inline fun CharSequence.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException(\"Empty char sequence can't be reduced.\")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last

character and applying [operation] from right to left\n * to each character with its index in the original char sequence and current accumulator value.\n * \n * Returns `null` if the char sequence is empty.\n * \n * @param [operation] function that takes the index of a character, the character itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\npublic inline fun CharSequence.reduceRightIndexedOrNull(operation: (index: Int, Char, acc: Char) -> Char): Char? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last character and applying [operation] from right to left\n * to each character and current accumulator value.\n * \n * Returns `null` if the char sequence is empty.\n * \n * @param [operation] function that takes a character and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun CharSequence.reduceRightOrNull(operation: (Char, acc: Char) -> Char): Char? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each character and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and a character, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R> CharSequence.runningFold(initial: R, operation: (acc: R, Char) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(length + 1).apply { add(initial) }\n var accumulator = initial\n for (element in this) {\n accumulator = operation(accumulator, element)\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each character, its index in the original char sequence and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of a character, current accumulator value\n * and the character itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\npublic inline fun <R> CharSequence.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(length + 1).apply { add(initial) }\n var accumulator = initial\n for (index in indices) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each character and current accumulator value that starts with the first character of this char sequence.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and a character, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\npublic inline fun CharSequence.runningReduce(operation: (acc: Char, Char) -> Char): List<Char> {\n if (isEmpty()) return emptyList()\n var accumulator = this[0]\n val result = ArrayList<Char>(length).apply { add(accumulator) }\n for (index in 1 until length) {\n accumulator = operation(accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each character, its index in the original char sequence and current accumulator value that starts with the first character of this char sequence.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of a character, current accumulator value\n * and the character itself, and calculates the next accumulator value.\n * \n * @sample

samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\npublic inline fun CharSequence.runningReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<Char>(length).apply { add(accumulator) }\n   for (index in 1 until length) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each character and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and a character, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> CharSequence.scan(initial: R, operation: (acc: R, Char) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each character, its index in the original char sequence and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of a character, current accumulator value\n * and the character itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> CharSequence.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {\n   return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun CharSequence.sumBy(selector: (Char) -> Int): Int {\n   var sum: Int = 0\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic inline fun CharSequence.sumByDouble(selector: (Char) -> Double): Double {\n   var sum: Double = 0.0\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.sumOf(selector: (Char) -> Double): Double {\n   var sum: Double = 0.toDouble()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.sumOf(selector: (Char) -> Int): Int {\n   var sum: Int = 0.toInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.sumOf(selector: (Char) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.sumOf(selector: (Char) -> UInt): UInt {\n

var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.sumOf(selector: (Char) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Splits this char sequence into a list of strings each not exceeding the given [size].\n * \n * The last string in the resulting list may have fewer characters than the given [size].\n * \n * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.\n * \n * @sample samples.text.Strings.chunked\n */\n@SinceKotlin(\"1.2\")\npublic fun CharSequence.chunked(size: Int): List<String> {\n   return windowed(size, size, partialWindows = true)\n}\n\n/**\n * Splits this char sequence into several char sequences each not exceeding the given [size]\n * and applies the given [transform] function to an each.\n * \n * @return list of results of the [transform] applied to an each char sequence.\n * \n * Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * The last char sequence may have fewer characters than the given [size].\n * \n * @param size the number of elements to take in each char sequence, must be positive and can be greater than the number of elements in this char sequence.\n * \n * @sample samples.text.Strings.chunkedTransform\n */\n@SinceKotlin(\"1.2\")\npublic fun <R> CharSequence.chunked(size: Int, transform: (CharSequence) -> R): List<R> {\n   return windowed(size, size, partialWindows = true, transform = transform)\n}\n\n/**\n * Splits this char sequence into a sequence of strings each not exceeding the given [size].\n * \n * The last string in the resulting sequence may have fewer characters than the given [size].\n * \n * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.\n * \n * @sample samples.collections.Collections.Transformations.chunked\n */\n@SinceKotlin(\"1.2\")\npublic fun CharSequence.chunkedSequence(size: Int): Sequence<String> {\n   return chunkedSequence(size) { it.toString() }\n}\n\n/**\n * Splits this char sequence into several char sequences each not exceeding the given [size]\n * and applies the given [transform] function to an each.\n * \n * @return sequence of results of the [transform] applied to an each char sequence.\n * \n * Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * The last char sequence may have fewer characters than the given [size].\n * \n * @param size the number of elements to take in each char sequence, must be positive and can be greater than the number of elements in this char sequence.\n * \n * @sample samples.text.Strings.chunkedTransformToSequence\n */\n@SinceKotlin(\"1.2\")\npublic fun <R> CharSequence.chunkedSequence(size: Int, transform: (CharSequence) -> R): Sequence<R> {\n   return windowedSequence(size, size, partialWindows = true, transform = transform)\n}\n\n/**\n * Splits the original char sequence into pair of char sequences,\n * where *first* char sequence contains characters for which [predicate] yielded `true`,\n * while *second* char sequence contains characters for which [predicate] yielded `false`.\n * \n * @sample samples.text.Strings.partition\n */\npublic inline fun CharSequence.partition(predicate: (Char) -> Boolean): Pair<CharSequence, CharSequence> {\n   val first = StringBuilder()\n   val second = StringBuilder()\n   for (element in this) {\n   if (predicate(element)) {\n   first.append(element)\n   } else {\n   second.append(element)\n   }\n   }\n   return Pair(first, second)\n}\n\n/**\n * Splits the original string into pair of strings,\n * where *first* string contains characters for which [predicate] yielded `true`,\n * while *second* string contains characters for which [predicate] yielded `false`.\n * \n * @sample samples.text.Strings.partition\n */\npublic inline fun String.partition(predicate: (Char) -> Boolean): Pair<String, String> {\n   val first = StringBuilder()\n   val second = StringBuilder()\n   for (element in this) {\n   if (predicate(element)) {\n   first.append(element)\n   } else {\n   second.append(element)\n   }\n   }\n   return Pair(first.toString(), second.toString())\n}\n\n/**\n * Returns a list of snapshots of the window of the given [size]\n * sliding along this char sequence with the given [step], where

each\n * snapshot is a string.\n * \n * Several last strings may have fewer characters than the given [size].\n * \n * Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.\n * @param size the number of elements to take in each window\n * @param step the number of elements to move the window forward by on an each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample samples.collections.Sequences.Transformations.takeWindows\n */\n@SinceKotlin(\"1.2\")\npublic fun CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): List<String> {\n   return windowed(size, step, partialWindows) { it.toString() }\n}\n\n/**\n * Returns a list of results of applying the given [transform] function to\n * an each char sequence representing a view over the window of the given [size]\n * sliding along this char sequence with the given [step].\n * \n * Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * Several last char sequences may have fewer characters than the given [size].\n * \n * Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.\n * @param size the number of elements to take in each window\n * @param step the number of elements to move the window forward by on an each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample samples.collections.Sequences.Transformations.averageWindows\n */\n@SinceKotlin(\"1.2\")\npublic fun <R> CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (CharSequence) -> R): List<R> {\n   checkWindowSizeStep(size, step)\n   val thisSize = this.length\n   val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1\n   val result = ArrayList<R>(resultCapacity)\n   var index = 0\n   while (index in 0 until thisSize) {\n   val end = index + size\n   val coercedEnd = if (end < 0 || end > thisSize) { if (partialWindows) thisSize else break } else end\n   result.add(transform(subSequence(index, coercedEnd)))\n   index += step\n   }\n   return result\n}\n\n/**\n * Returns a sequence of snapshots of the window of the given [size]\n * sliding along this char sequence with the given [step], where each\n * snapshot is a string.\n * \n * Several last strings may have fewer characters than the given [size].\n * \n * Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.\n * @param size the number of elements to take in each window\n * @param step the number of elements to move the window forward by on an each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample samples.collections.Sequences.Transformations.takeWindows\n */\n@SinceKotlin(\"1.2\")\npublic fun CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false): Sequence<String> {\n   return windowedSequence(size, step, partialWindows) { it.toString() }\n}\n\n/**\n * Returns a sequence of results of applying the given [transform] function to\n * an each char sequence representing a view over the window of the given [size]\n * sliding along this char sequence with the given [step].\n * \n * Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * Several last char sequences may have fewer characters than the given [size].\n * \n * Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.\n * @param size the number of elements to take in each window\n * @param step the number of elements to move the window forward by on an each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample samples.collections.Sequences.Transformations.averageWindows\n */\n@SinceKotlin(\"1.2\")\npublic fun <R> CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (CharSequence) -> R): Sequence<R> {\n   checkWindowSizeStep(size, step)\n   val windows = (if (partialWindows) indices else 0 until length - size + 1) step step\n   return windows.asSequence().map { index ->\n   val end = index + size\n   val coercedEnd = if (end < 0 || end > length) length else end\n   transform(subSequence(index, coercedEnd))\n   }\n}\n\n/**\n * Returns a list of pairs built from the characters of `this` and the [other] char sequences with the same index\n * The returned list has length of the shortest char

sequence.\n * \n * @sample samples.text.Strings.zip\n */\npublic infix fun CharSequence.zip(other: CharSequence): List<Pair<Char, Char>> {\n   return zip(other) { c1, c2 -> c1 to c2 }\n}\n\n/**\n * Returns a list of values built from the characters of `this` and the [other] char sequences with the same index\n * using the provided [transform] function applied to each pair of characters.\n * The returned list has length of the shortest char sequence.\n * \n * @sample samples.text.Strings.zipWithTransform\n */\npublic inline fun <V> CharSequence.zip(other: CharSequence, transform: (a: Char, b: Char) -> V): List<V> {\n   val length = minOf(this.length, other.length)\n   val list = ArrayList<V>(length)\n   for (i in 0 until length) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of pairs of each two adjacent characters in this char sequence.\n * \n * The returned list is empty if this char sequence contains less than two characters.\n * \n * @sample samples.collections.Collections.Transformations.zipWithNext\n */\n@SinceKotlin(\"1.2\")\npublic fun CharSequence.zipWithNext(): List<Pair<Char, Char>> {\n   return zipWithNext { a, b -> a to b }\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to an each pair of two adjacent characters in this char sequence.\n * \n * The returned list is empty if this char sequence contains less than two characters.\n * \n * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas\n */\n@SinceKotlin(\"1.2\")\npublic inline fun <R> CharSequence.zipWithNext(transform: (a: Char, b: Char) -> R): List<R> {\n   val size = length - 1\n   if (size < 1) return emptyList()\n   val result = ArrayList<R>(size)\n   for (index in 0 until size) {\n   result.add(transform(this[index], this[index + 1]))\n   }\n   return result\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original char sequence returning its characters when being iterated.\n */\npublic fun CharSequence.asIterable(): Iterable<Char> {\n   if (this is String && isEmpty()) return emptyList()\n   return Iterable { this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original char sequence returning its characters when being iterated.\n */\npublic fun CharSequence.asSequence(): Sequence<Char> {\n   if (this is String && isEmpty()) return emptySequence()\n   return Sequence { this.iterator() }\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"StringsKt\")\n\npackage kotlin.text\n\nimport kotlin.contracts.contract\nimport kotlin.jvm.JvmName\n\n/**\n * Returns a copy of this string converted to upper case using the rules of the default locale.\n */\n@Deprecated(\"Use uppercase() instead.\", ReplaceWith(\"uppercase()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic expect fun String.toUpperCase(): String\n\n/**\n * Returns a copy of this string converted to upper case using Unicode mapping rules of the invariant locale.\n *\n * This function supports one-to-many and many-to-one character mapping,\n * thus the length of the returned string can be different from the length of the original string.\n *\n * @sample samples.text.Strings.uppercase\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun String.uppercase(): String\n\n/**\n * Returns a copy of this string converted to lower case using the rules of the default locale.\n */\n@Deprecated(\"Use lowercase() instead.\", ReplaceWith(\"lowercase()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic expect fun String.toLowerCase(): String\n\n/**\n * Returns a copy of this string converted to lower case using Unicode mapping rules of the invariant locale.\n *\n * This function supports one-to-many and many-to-one character mapping,\n * thus the length of the returned string can be different from the length of the original string.\n *\n * @sample samples.text.Strings.lowercase\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun String.lowercase(): String\n\n/**\n * Returns a copy of this string having its first letter titlecased using the rules of the default locale,\n * or the original string if it's empty or already starts with a title case letter.\n *\n * The title case of a character is usually the same as its upper case with several exceptions.\n * The particular list of characters with the special title case form depends on the underlying platform.\n *\n * @sample samples.text.Strings.capitalize\n */\n@Deprecated(\"Use replaceFirstChar instead.\", ReplaceWith(\"replaceFirstChar { if (it.isLowerCase()) it.titlecase() else it.toString() }\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic expect fun String.capitalize(): String\n\n/**\n * Returns a copy of this string having its first letter lowercased using the rules of

the default locale,\n * or the original string if it's empty or already starts with a lower case letter.\n *\n * @sample samples.text.Strings.decapitalize\n */\n@Deprecated(\"Use replaceFirstChar instead.\", ReplaceWith(\"replaceFirstChar { it.lowercase() }\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic expect fun String.decapitalize(): String\n\n/**\n * Returns a sub sequence of this char sequence having leading and trailing characters matching the [predicate] removed.\n */\npublic inline fun CharSequence.trim(predicate: (Char) -> Boolean): CharSequence {\n   var startIndex = 0\n   var endIndex = length - 1\n   var startFound = false\n\n   while (startIndex <= endIndex) {\n   val index = if (!startFound) startIndex else endIndex\n   val match = predicate(this[index])\n\n   if (!startFound) {\n   if (!match)\n   startFound = true\n   else\n   startIndex += 1\n   } else {\n   if (!match)\n   break\n   else\n   endIndex -= 1\n   }\n   }\n\n   return subSequence(startIndex, endIndex + 1)\n}\n\n/**\n * Returns a string having leading and trailing characters matching the [predicate] removed.\n */\npublic inline fun String.trim(predicate: (Char) -> Boolean): String =\n   (this as CharSequence).trim(predicate).toString()\n\n/**\n * Returns a sub sequence of this char sequence having leading characters matching the [predicate] removed.\n */\npublic inline fun CharSequence.trimStart(predicate: (Char) -> Boolean): CharSequence {\n   for (index in this.indices)\n   if (!predicate(this[index]))\n   return subSequence(index, length)\n\n   return \"\"\n}\n\n/**\n * Returns a string having leading characters matching the [predicate] removed.\n */\npublic inline fun String.trimStart(predicate: (Char) -> Boolean): String =\n   (this as CharSequence).trimStart(predicate).toString()\n\n/**\n * Returns a sub sequence of this char sequence having trailing characters matching the [predicate] removed.\n */\npublic inline fun CharSequence.trimEnd(predicate: (Char) -> Boolean): CharSequence {\n   for (index in this.indices.reversed())\n   if (!predicate(this[index]))\n   return subSequence(0, index + 1)\n\n   return \"\"\n}\n\n/**\n * Returns a string having trailing characters matching the [predicate] removed.\n */\npublic inline fun String.trimEnd(predicate: (Char) -> Boolean): String =\n   (this as CharSequence).trimEnd(predicate).toString()\n\n/**\n * Returns a sub sequence of this char sequence having leading and trailing characters from the [chars] array removed.\n */\npublic fun CharSequence.trim(vararg chars: Char): CharSequence = trim { it in chars }\n\n/**\n * Returns a string having leading and trailing characters from the [chars] array removed.\n */\npublic fun String.trim(vararg chars: Char): String = trim { it in chars }\n\n/**\n * Returns a sub sequence of this char sequence having leading characters from the [chars] array removed.\n */\npublic fun CharSequence.trimStart(vararg chars: Char): CharSequence = trimStart { it in chars }\n\n/**\n * Returns a string having leading characters from the [chars] array removed.\n */\npublic fun String.trimStart(vararg chars: Char): String = trimStart { it in chars }\n\n/**\n * Returns a sub sequence of this char sequence having trailing characters from the [chars] array removed.\n */\npublic fun CharSequence.trimEnd(vararg chars: Char): CharSequence = trimEnd { it in chars }\n\n/**\n * Returns a string having trailing characters from the [chars] array removed.\n */\npublic fun String.trimEnd(vararg chars: Char): String = trimEnd { it in chars }\n\n/**\n * Returns a sub sequence of this char sequence having leading and trailing whitespace removed.\n */\npublic fun CharSequence.trim(): CharSequence = trim(Char::isWhitespace)\n\n/**\n * Returns a string having leading and trailing whitespace removed.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.trim(): String = (this as CharSequence).trim().toString()\n\n/**\n * Returns a sub sequence of this char sequence having leading whitespace removed.\n */\npublic fun CharSequence.trimStart(): CharSequence = trimStart(Char::isWhitespace)\n\n/**\n * Returns a string having leading whitespace removed.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.trimStart(): String = (this as CharSequence).trimStart().toString()\n\n/**\n * Returns a sub sequence of this char sequence having trailing whitespace removed.\n */\npublic fun CharSequence.trimEnd(): CharSequence = trimEnd(Char::isWhitespace)\n\n/**\n * Returns a string having trailing whitespace removed.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.trimEnd(): String = (this as CharSequence).trimEnd().toString()\n\n/**\n * Returns a char sequence with content of this char sequence padded at the beginning\n * to the specified [length] with the specified character or space.\n *\n * @param length the desired string length.\n * @param padChar the character to pad string with, if it has length less than the [length] specified. Space is used by default.\n * @return Returns a char sequence of length at least [length] consisting of `this` char sequence prepended with [padChar] as many times\n * as are necessary to reach that length.\n * @sample samples.text.Strings.padStart\n */\npublic fun CharSequence.padStart(length: Int, padChar: Char = ' '):

CharSequence {\n   if (length < 0)\n   throw IllegalArgumentException(\"Desired length $length is less than zero.\")\n   if (length <= this.length)\n   return this.subSequence(0, this.length)\n\n   val sb = StringBuilder(length)\n   for (i in 1..(length - this.length))\n   sb.append(padChar)\n   sb.append(this)\n   return sb\n}\n\n/**\n * Pads the string to the specified [length] at the beginning with the specified character or space.\n *\n * @param length the desired string length.\n * @param padChar the character to pad string with, if it has length less than the [length] specified. Space is used by default.\n * @return Returns a string of length at least [length] consisting of `this` string prepended with [padChar] as many times\n * as are necessary to reach that length.\n * @sample samples.text.Strings.padStart\n */\npublic fun String.padStart(length: Int, padChar: Char = ' '): String =\n(this as CharSequence).padStart(length, padChar).toString()\n\n/**\n * Returns a char sequence with content of this char sequence padded at the end\n * to the specified [length] with the specified character or space.\n *\n * @param length the desired string length.\n * @param padChar the character to pad string with, if it has length less than the [length] specified. Space is used by default.\n * @return Returns a char sequence of length at least [length] consisting of `this` char sequence appended with [padChar] as many times\n * as are necessary to reach that length.\n * @sample samples.text.Strings.padEnd\n */\npublic fun CharSequence.padEnd(length: Int, padChar: Char = ' '): CharSequence {\n   if (length < 0)\n   throw IllegalArgumentException(\"Desired length $length is less than zero.\")\n   if (length <= this.length)\n   return this.subSequence(0, this.length)\n\n   val sb = StringBuilder(length)\n   sb.append(this)\n   for (i in 1..(length - this.length))\n   sb.append(padChar)\n   return sb\n}\n\n/**\n * Pads the string to the specified [length] at the end with the specified character or space.\n *\n * @param length the desired string length.\n * @param padChar the character to pad string with, if it has length less than the [length] specified. Space is used by default.\n * @return Returns a string of length at least [length] consisting of `this` string appended with [padChar] as many times\n * as are necessary to reach that length.\n * @sample samples.text.Strings.padEnd\n */\npublic fun String.padEnd(length: Int, padChar: Char = ' '): String =\n(this as CharSequence).padEnd(length, padChar).toString()\n\n/**\n * Returns `true` if this nullable char sequence is either `null` or empty.\n *\n * @sample samples.text.Strings.stringIsNullOrEmpty\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence?.isNullOrEmpty(): Boolean {\n   contract {\n returns(false) implies (this@isNullOrEmpty != null)\n   }\n\n   return this == null || this.length == 0\n}\n\n/**\n * Returns `true` if this char sequence is empty (contains no characters).\n *\n * @sample samples.text.Strings.stringIsEmpty\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.isEmpty(): Boolean = length == 0\n\n/**\n * Returns `true` if this char sequence is not empty.\n *\n * @sample samples.text.Strings.stringIsNotEmpty\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.isNotEmpty(): Boolean = length > 0\n\n// implemented differently in JVM and JS\n//public fun String.isBlank(): Boolean = length() == 0 || all { it.isWhitespace() }\n\n/**\n * Returns `true` if this char sequence is not empty and contains some characters except of whitespace characters.\n *\n * @sample samples.text.Strings.stringIsNotBlank\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.isNotBlank(): Boolean = !isBlank()\n\n/**\n * Returns `true` if this nullable char sequence is either `null` or empty or consists solely of whitespace characters.\n *\n * @sample samples.text.Strings.stringIsNullOrBlank\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence?.isNullOrBlank(): Boolean {\n   contract {\n   returns(false) implies (this@isNullOrBlank != null)\n   }\n\n   return this == null || this.isBlank()\n}\n\n/**\n * Iterator for characters of the given char sequence.\n */\npublic operator fun CharSequence.iterator(): CharIterator = object : CharIterator() {\n   private var index = 0\n\n   public override fun nextChar(): Char = get(index++)\n\n   public override fun hasNext(): Boolean = index < length\n}\n\n/** Returns the string if it is not `null`, or the empty string otherwise.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String?.orEmpty(): String = this ?: \"\"\n\n/**\n * Returns this char sequence if it's not empty\n * or the result of calling [defaultValue] function if the char sequence is empty.\n *\n * @sample samples.text.Strings.stringIfEmpty\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : CharSequence, C : R =\n   if (isEmpty()) defaultValue() else this\n\n/**\n * Returns this char sequence if it is not empty and doesn't consist solely of whitespace characters,\n * or the result of calling

[defaultValue] function otherwise.\n *\n * @sample samples.text.Strings.stringIfBlank\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun <C, R> C.ifBlank(defaultValue: () -> R): R where C : CharSequence, C : R =\n   if (isBlank()) defaultValue() else this\n\n/**\n * Returns the range of valid character indices for this char sequence.\n */\npublic val CharSequence.indices: IntRange\n   get() = 0..length - 1\n\n/**\n * Returns the index of the last character in the char sequence or -1 if it is empty.\n */\npublic val CharSequence.lastIndex: Int\n   get() = this.length - 1\n\n/**\n * Returns `true` if this CharSequence has Unicode surrogate pair at the specified [index].\n */\npublic fun CharSequence.hasSurrogatePairAt(index: Int): Boolean {\n return index in 0..length - 2\n   && this[index].isHighSurrogate()\n   && this[index + 1].isLowSurrogate()\n}\n\n/**\n * Returns a substring specified by the given [range] of indices.\n */\npublic fun String.substring(range: IntRange): String = substring(range.start, range.endInclusive + 1)\n\n/**\n * Returns a subsequence of this char sequence specified by the given [range] of indices.\n */\npublic fun CharSequence.subSequence(range: IntRange): CharSequence = subSequence(range.start, range.endInclusive + 1)\n\n/**\n * Returns a subsequence of this char sequence.\n *\n * This extension is chosen only for invocation with old-named parameters.\n * Replace parameter names with the same as those of [CharSequence.subSequence].\n */\n@kotlin.internal.InlineOnly\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\") // false warning\n@Deprecated(\"Use parameters named startIndex and endIndex.\", ReplaceWith(\"subSequence(startIndex = start, endIndex = end)\"))\npublic inline fun String.subSequence(start: Int, end: Int): CharSequence = subSequence(start, end)\n\n/**\n * Returns a substring of chars from a range of this char sequence starting at the [startIndex] and ending right before the [endIndex].\n *\n * @param startIndex the start index (inclusive).\n * @param endIndex the end index (exclusive). If not specified, the length of the char sequence is used.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.substring(startIndex: Int, endIndex: Int = length): String = subSequence(startIndex, endIndex).toString()\n\n/**\n * Returns a substring of chars at indices from the specified [range] of this char sequence.\n */\npublic fun CharSequence.substring(range: IntRange): String = subSequence(range.start, range.endInclusive + 1).toString()\n\n/**\n * Returns a substring before the first occurrence of [delimiter].\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.substringBefore(delimiter: Char, missingDelimiterValue: String = this): String {\n   val index = indexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else substring(0, index)\n}\n\n/**\n * Returns a substring before the first occurrence of [delimiter].\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.substringBefore(delimiter: String, missingDelimiterValue: String = this): String {\n   val index = indexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else substring(0, index)\n}\n\n/**\n * Returns a substring after the first occurrence of [delimiter].\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.substringAfter(delimiter: Char, missingDelimiterValue: String = this): String {\n   val index = indexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else substring(index + 1, length)\n}\n\n/**\n * Returns a substring after the first occurrence of [delimiter].\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.substringAfter(delimiter: String, missingDelimiterValue: String = this): String {\n   val index = indexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)\n}\n\n/**\n * Returns a substring before the last occurrence of [delimiter].\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.substringBeforeLast(delimiter: Char, missingDelimiterValue: String = this): String {\n   val index = lastIndexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else substring(0, index)\n}\n\n/**\n * Returns a substring before the last occurrence of [delimiter].\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.substringBeforeLast(delimiter: String, missingDelimiterValue: String = this): String {\n   val index = lastIndexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else substring(0, index)\n}\n\n/**\n * Returns a substring after the last occurrence of [delimiter].\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.substringAfterLast(delimiter: Char, missingDelimiterValue: String = this):

String {\n   val index = lastIndexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else substring(index + 1, length)\n}\n\n/**\n * Returns a substring after the last occurrence of [delimiter].\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.substringAfterLast(delimiter: String, missingDelimiterValue: String = this): String {\n   val index = lastIndexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)\n}\n\n/**\n * Returns a char sequence with content of this char sequence where its part at the given range\n * is replaced with the [replacement] char sequence.\n * @param startIndex the index of the first character to be replaced.\n * @param endIndex the index of the first character after the replacement to keep in the string.\n */\npublic fun CharSequence.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): CharSequence {\n   if (endIndex < startIndex)\n   throw IndexOutOfBoundsException(\"End index ($endIndex) is less than start index ($startIndex).\")\n   val sb = StringBuilder()\n   sb.appendRange(this, 0, startIndex)\n   sb.append(replacement)\n   sb.appendRange(this, endIndex, length)\n   return sb\n}\n\n/**\n * Replaces the part of the string at the given range with the [replacement] char sequence.\n * @param startIndex the index of the first character to be replaced.\n * @param endIndex the index of the first character after the replacement to keep in the string.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): String =\n   (this as CharSequence).replaceRange(startIndex, endIndex, replacement).toString()\n\n/**\n * Returns a char sequence with content of this char sequence where its part at the given [range]\n * is replaced with the [replacement] char sequence.\n *\n * The end index of the [range] is included in the part to be replaced.\n */\npublic fun CharSequence.replaceRange(range: IntRange, replacement: CharSequence): CharSequence =\n   replaceRange(range.start, range.endInclusive + 1, replacement)\n\n/**\n * Replace the part of string at the given [range] with the [replacement] string.\n *\n * The end index of the [range] is included in the part to be replaced.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.replaceRange(range: IntRange, replacement: CharSequence): String =\n   (this as CharSequence).replaceRange(range, replacement).toString()\n\n/**\n * Returns a char sequence with content of this char sequence where its part at the given range is removed.\n *\n * @param startIndex the index of the first character to be removed.\n * @param endIndex the index of the first character after the removed part to keep in the string.\n *\n * [endIndex] is not included in the removed part.\n */\npublic fun CharSequence.removeRange(startIndex: Int, endIndex: Int): CharSequence {\n   if (endIndex < startIndex)\n   throw IndexOutOfBoundsException(\"End index ($endIndex) is less than start index ($startIndex).\")\n\n   if (endIndex == startIndex)\n   return this.subSequence(0, length)\n\n   val sb = StringBuilder(length - (endIndex - startIndex))\n   sb.appendRange(this, 0, startIndex)\n   sb.appendRange(this, endIndex, length)\n   return sb\n}\n\n/**\n * Removes the part of a string at a given range.\n * @param startIndex the index of the first character to be removed.\n * @param endIndex the index of the first character after the removed part to keep in the string.\n *\n * [endIndex] is not included in the removed part.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.removeRange(startIndex: Int, endIndex: Int): String =\n   (this as CharSequence).removeRange(startIndex, endIndex).toString()\n\n/**\n * Returns a char sequence with content of this char sequence where its part at the given [range] is removed.\n *\n * The end index of the [range] is included in the removed part.\n */\npublic fun CharSequence.removeRange(range: IntRange): CharSequence = removeRange(range.start, range.endInclusive + 1)\n\n/**\n * Removes the part of a string at the given [range].\n *\n * The end index of the [range] is included in the removed part.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.removeRange(range: IntRange): String =\n   (this as CharSequence).removeRange(range).toString()\n\n/**\n * If this char sequence starts with the given [prefix], returns a new char sequence\n * with the prefix removed. Otherwise, returns a new char sequence with the same characters.\n */\npublic fun CharSequence.removePrefix(prefix: CharSequence): CharSequence {\n   if (startsWith(prefix)) {\n   return subSequence(prefix.length, length)\n   }\n   return subSequence(0, length)\n}\n\n/**\n * If this string starts with the given [prefix], returns a copy of this string\n * with the prefix removed. Otherwise, returns this string.\n */\npublic fun String.removePrefix(prefix: CharSequence): String {\n   if (startsWith(prefix)) {\n   return substring(prefix.length)\n   }\n   return this\n}\n\n/**\n * If this char sequence ends with the given [suffix], returns a new char sequence\n * with the suffix removed. Otherwise, returns a new char sequence with the same

characters.\n */\npublic fun CharSequence.removeSuffix(suffix: CharSequence): CharSequence {\n   if (endsWith(suffix)) {\n   return subSequence(0, length - suffix.length)\n   }\n   return subSequence(0, length)\n}\n\n/**\n * If this string ends with the given [suffix], returns a copy of this string\n * with the suffix removed. Otherwise, returns this string.\n */\npublic fun String.removeSuffix(suffix: CharSequence): String {\n   if (endsWith(suffix)) {\n   return substring(0, length - suffix.length)\n   }\n   return this\n}\n\n/**\n * When this char sequence starts with the given [prefix] and ends with the given [suffix],\n * returns a new char sequence having both the given [prefix] and [suffix] removed.\n * Otherwise returns a new char sequence with the same characters.\n */\npublic fun CharSequence.removeSurrounding(prefix: CharSequence, suffix: CharSequence): CharSequence {\n   if ((length >= prefix.length + suffix.length) && startsWith(prefix) && endsWith(suffix)) {\n   return subSequence(prefix.length, length - suffix.length)\n   }\n   return subSequence(0, length)\n}\n\n/**\n * Removes from a string both the given [prefix] and [suffix] if and only if\n * it starts with the [prefix] and ends with the [suffix].\n * Otherwise returns this string unchanged.\n */\npublic fun String.removeSurrounding(prefix: CharSequence, suffix: CharSequence): String {\n   if ((length >= prefix.length + suffix.length) && startsWith(prefix) && endsWith(suffix)) {\n   return substring(prefix.length, length - suffix.length)\n   }\n   return this\n}\n\n/**\n * When this char sequence starts with and ends with the given [delimiter],\n * returns a new char sequence having this [delimiter] removed both from the start and end.\n * Otherwise returns a new char sequence with the same characters.\n */\npublic fun CharSequence.removeSurrounding(delimiter: CharSequence): CharSequence = removeSurrounding(delimiter, delimiter)\n\n/**\n * Removes the given [delimiter] string from both the start and the end of this string\n * if and only if it starts with and ends with the [delimiter].\n * Otherwise returns this string unchanged.\n */\npublic fun String.removeSurrounding(delimiter: CharSequence): String = removeSurrounding(delimiter, delimiter)\n\n/**\n * Replace part of string before the first occurrence of given delimiter with the [replacement] string.\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.replaceBefore(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {\n   val index = indexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)\n}\n\n/**\n * Replace part of string before the first occurrence of given delimiter with the [replacement] string.\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.replaceBefore(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {\n   val index = indexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)\n}\n\n/**\n * Replace part of string after the first occurrence of given delimiter with the [replacement] string.\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.replaceAfter(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {\n   val index = indexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else replaceRange(index + 1, length, replacement)\n}\n\n/**\n * Replace part of string after the first occurrence of given delimiter with the [replacement] string.\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.replaceAfter(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {\n   val index = indexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else replaceRange(index + delimiter.length, length, replacement)\n}\n\n/**\n * Replace part of string after the last occurrence of given delimiter with the [replacement] string.\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.replaceAfterLast(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {\n   val index = lastIndexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else replaceRange(index + delimiter.length, length, replacement)\n}\n\n/**\n * Replace part of string after the last occurrence of given delimiter with the [replacement] string.\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.replaceAfterLast(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {\n   val index = lastIndexOf(delimiter)\n   return if (index == -1) missingDelimiterValue else replaceRange(index + 1, length, replacement)\n}\n\n/**\n * Replace part of string before the last occurrence of given delimiter with the [replacement] string.\n * If the string

does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.replaceBeforeLast(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {\n val index = lastIndexOf(delimiter)\n return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)\n}\n\n/**\n * Replace part of string before the last occurrence of given delimiter with the [replacement] string.\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.\n */\npublic fun String.replaceBeforeLast(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {\n val index = lastIndexOf(delimiter)\n return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)\n}\n\n\n// public fun String.replace(oldChar: Char, newChar: Char, ignoreCase: Boolean): String // JVM- and JS-specific\n// public fun String.replace(oldValue: String, newValue: String, ignoreCase: Boolean): String // JVM- and JS-specific\n\n/**\n * Returns a new string obtained by replacing each substring of this char sequence that matches the given regular expression\n * with the given [replacement].\n *\n * The [replacement] can consist of any combination of literal text and $-substitutions. To treat the replacement string\n * literally escape it with the [kotlin.text.Regex.Companion.escapeReplacement] method.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.replace(regex: Regex, replacement: String): String = regex.replace(this, replacement)\n\n/**\n * Returns a new string obtained by replacing each substring of this char sequence that matches the given regular expression\n * with the result of the given function [transform] that takes [MatchResult] and returns a string to be used as a\n * replacement for that match.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.replace(regex: Regex, noinline transform: (MatchResult) -> CharSequence): String =\n regex.replace(this, transform)\n\n/**\n * Replaces the first occurrence of the given regular expression [regex] in this char sequence with specified [replacement] expression.\n *\n * @param replacement A replacement expression that can include substitutions. See [Regex.replaceFirst] for details.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.replaceFirst(regex: Regex, replacement: String): String = regex.replaceFirst(this, replacement)\n\n/**\n * Returns a copy of this string having its first character replaced with the result of the specified [transform],\n * or the original string if it's empty.\n *\n * @param transform function that takes the first character and returns the result of the transform applied to the character.\n *\n * @sample samples.text.Strings.replaceFirstChar\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@JvmName(\"replaceFirstCharWithChar\")\n@kotlin.internal.InlineOnly\npublic inline fun String.replaceFirstChar(transform: (Char) -> Char): String {\n return if (isNotEmpty()) transform(this[0]) + substring(1) else this\n}\n\n/**\n * Returns a copy of this string having its first character replaced with the result of the specified [transform],\n * or the original string if it's empty.\n *\n * @param transform function that takes the first character and returns the result of the transform applied to the character.\n *\n * @sample samples.text.Strings.replaceFirstChar\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@JvmName(\"replaceFirstCharWithCharSequence\")\n@kotlin.internal.InlineOnly\npublic inline fun String.replaceFirstChar(transform: (Char) -> CharSequence): String {\n return if (isNotEmpty()) transform(this[0]).toString() + substring(1) else this\n}\n\n\n/**\n * Returns `true` if this char sequence matches the given regular expression.\n */\n@kotlin.internal.InlineOnly\npublic inline infix fun CharSequence.matches(regex: Regex): Boolean = regex.matches(this)\n\n/**\n * Implementation of [regionMatches] for CharSequences.\n * Invoked when it's already known that arguments are not Strings, so that no additional type checks are performed.\n */\ninternal fun CharSequence.regionMatchesImpl(thisOffset: Int, other: CharSequence, otherOffset: Int, length: Int, ignoreCase: Boolean): Boolean {\n if ((otherOffset < 0) || (thisOffset < 0) || (thisOffset > this.length - length) || (otherOffset > other.length - length)) {\n   return false\n }\n\n for (index in 0 until length) {\n   if (!this[thisOffset + index].equals(other[otherOffset + index], ignoreCase))\n   return false\n }\n return true\n}\n\n/**\n * Returns `true` if this char sequence starts with the specified character.\n */\npublic fun CharSequence.startsWith(char: Char, ignoreCase: Boolean = false): Boolean =\n this.length > 0 && this[0].equals(char, ignoreCase)\n\n/**\n * Returns `true` if this char sequence ends with the specified character.\n

```
*/\npublic fun CharSequence.endsWith(char: Char, ignoreCase: Boolean = false): Boolean =\n    this.length > 0 && this[lastIndex].equals(char, ignoreCase)\n\n/**\n * Returns `true` if this char sequence starts with the specified prefix.\n */\npublic fun CharSequence.startsWith(prefix: CharSequence, ignoreCase: Boolean = false): Boolean {\n    if (!ignoreCase && this is String && prefix is String)\n        return this.startsWith(prefix)\n    else\n        return regionMatchesImpl(0, prefix, 0, prefix.length, ignoreCase)\n}\n\n/**\n * Returns `true` if a substring of this char sequence starting at the specified offset [startIndex] starts with the specified prefix.\n */\npublic fun CharSequence.startsWith(prefix: CharSequence, startIndex: Int, ignoreCase: Boolean = false): Boolean {\n    if (!ignoreCase && this is String && prefix is String)\n        return this.startsWith(prefix, startIndex)\n    else\n        return regionMatchesImpl(startIndex, prefix, 0, prefix.length, ignoreCase)\n}\n\n/**\n * Returns `true` if this char sequence ends with the specified suffix.\n */\npublic fun CharSequence.endsWith(suffix: CharSequence, ignoreCase: Boolean = false): Boolean {\n    if (!ignoreCase && this is String && suffix is String)\n        return this.endsWith(suffix)\n    else\n        return regionMatchesImpl(length - suffix.length, suffix, 0, suffix.length, ignoreCase)\n}\n\n\n// common prefix and suffix\n\n/**\n * Returns the longest string `prefix` such that this char sequence and [other] char sequence both start with this prefix,\n * taking care not to split surrogate pairs.\n * If this and [other] have no common prefix, returns the empty string.\n\n * @param ignoreCase `true` to ignore character case when matching a character. By default `false`.\n * @sample samples.text.Strings.commonPrefixWith\n */\npublic fun CharSequence.commonPrefixWith(other: CharSequence, ignoreCase: Boolean = false): String {\n    val shortestLength = minOf(this.length, other.length)\n\n    var i = 0\n    while (i < shortestLength && this[i].equals(other[i], ignoreCase = ignoreCase)) {\n        i++\n    }\n    if (this.hasSurrogatePairAt(i - 1) || other.hasSurrogatePairAt(i - 1)) {\n        i--\n    }\n    return subSequence(0, i).toString()\n}\n\n/**\n * Returns the longest string `suffix` such that this char sequence and [other] char sequence both end with this suffix,\n * taking care not to split surrogate pairs.\n * If this and [other] have no common suffix, returns the empty string.\n\n * @param ignoreCase `true` to ignore character case when matching a character. By default `false`.\n * @sample samples.text.Strings.commonSuffixWith\n */\npublic fun CharSequence.commonSuffixWith(other: CharSequence, ignoreCase: Boolean = false): String {\n    val thisLength = this.length\n    val otherLength = other.length\n    val shortestLength = minOf(thisLength, otherLength)\n\n    var i = 0\n    while (i < shortestLength && this[thisLength - i - 1].equals(other[otherLength - i - 1], ignoreCase = ignoreCase)) {\n        i++\n    }\n    if (this.hasSurrogatePairAt(thisLength - i - 1) || other.hasSurrogatePairAt(otherLength - i - 1)) {\n        i--\n    }\n    return subSequence(thisLength - i, thisLength).toString()\n}\n\n\n// indexOfAny()\n\n/**\n * Finds the index of the first occurrence of any of the specified [chars] in this char sequence,\n * starting from the specified [startIndex] and optionally ignoring the case.\n *\n * @param ignoreCase `true` to ignore character case when matching a character. By default `false`.\n * @return An index of the first occurrence of matched character from [chars] or -1 if none of [chars] are found.\n *\n */\npublic fun CharSequence.indexOfAny(chars: CharArray, startIndex: Int = 0, ignoreCase: Boolean = false): Int {\n    if (!ignoreCase && chars.size == 1 && this is String) {\n        val char = chars.single()\n        return nativeIndexOf(char, startIndex)\n    }\n\n    for (index in startIndex.coerceAtLeast(0)..lastIndex) {\n        val charAtIndex = get(index)\n        if (chars.any { it.equals(charAtIndex, ignoreCase) })\n            return index\n    }\n    return -1\n}\n\n/**\n * Finds the index of the last occurrence of any of the specified [chars] in this char sequence,\n * starting from the specified [startIndex] and optionally ignoring the case.\n *\n * @param startIndex The index of character to start searching at. The search proceeds backward toward the beginning of the string.\n * @param ignoreCase `true` to ignore character case when matching a character. By default `false`.\n * @return An index of the last occurrence of matched character from [chars] or -1 if none of [chars] are found.\n *\n */\npublic fun CharSequence.lastIndexOfAny(chars: CharArray, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int {\n    if (!ignoreCase && chars.size == 1 && this is String) {\n        val char = chars.single()\n        return nativeLastIndexOf(char, startIndex)\n    }\n\n    for (index in startIndex.coerceAtMost(lastIndex) downTo 0) {\n        val charAtIndex = get(index)\n        if (chars.any { it.equals(charAtIndex, ignoreCase) })\n            return index\n    }\n\n    return -1\n}\n\n\nprivate fun CharSequence.indexOf(other: CharSequence, startIndex: Int, endIndex: Int, ignoreCase: Boolean, last: Boolean = false): Int {\n    val indices = if (!last)\n        startIndex.coerceAtLeast(0)..endIndex.coerceAtMost(length)\n    else\n        startIndex.coerceAtMost(lastIndex)
```

downTo endIndex.coerceAtLeast(0)\n\n   if (this is String && other is String) { // smart cast\n   for (index in indices) {\n   if (other.regionMatches(0, this, index, other.length, ignoreCase))\n   return index\n   }\n   } else {\n   for (index in indices) {\n   if (other.regionMatchesImpl(0, this, index, other.length, ignoreCase))\n   return index\n   }\n   }\n   return -1\n}\n\nprivate fun CharSequence.findAnyOf(strings: Collection<String>, startIndex: Int, ignoreCase: Boolean, last: Boolean): Pair<Int, String>? {\n   if (!ignoreCase && strings.size == 1) {\n   val string = strings.single()\n   val index = if (!last) indexOf(string, startIndex) else lastIndexOf(string, startIndex)\n   return if (index < 0) null else index to string\n   }\n\n   val indices = if (!last) startIndex.coerceAtLeast(0)..length else startIndex.coerceAtMost(lastIndex) downTo 0\n\n   if (this is String) {\n   for (index in indices) {\n   val matchingString = strings.firstOrNull { it.regionMatches(0, this, index, it.length, ignoreCase) }\n   if (matchingString != null)\n   return index to matchingString\n   }\n   } else {\n   for (index in indices) {\n   val matchingString = strings.firstOrNull { it.regionMatchesImpl(0, this, index, it.length, ignoreCase) }\n   if (matchingString != null)\n   return index to matchingString\n   }\n   }\n\n   return null\n}\n\n/**\n * Finds the first occurrence of any of the specified [strings] in this char sequence,\n * starting from the specified [startIndex] and optionally ignoring the case.\n *\n * @param ignoreCase `true` to ignore character case when matching a string. By default `false`.\n * @return A pair of an index of the first occurrence of matched string from [strings] and the string matched\n * or `null` if none of [strings] are found.\n *\n * To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from\n * the beginning to the end of this string, and finds at each position the first element in [strings]\n * that matches this string at that position.\n */\npublic fun CharSequence.findAnyOf(strings: Collection<String>, startIndex: Int = 0, ignoreCase: Boolean = false): Pair<Int, String>? =\n   findAnyOf(strings, startIndex, ignoreCase, last = false)\n\n/**\n * Finds the last occurrence of any of the specified [strings] in this char sequence,\n * starting from the specified [startIndex] and optionally ignoring the case.\n *\n * @param startIndex The index of character to start searching at. The search proceeds backward toward the beginning of the string.\n * @param ignoreCase `true` to ignore character case when matching a string. By default `false`.\n * @return A pair of an index of the last occurrence of matched string from [strings] and the string matched or `null` if none of [strings] are found.\n *\n * To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from\n * the end toward the beginning of this string, and finds at each position the first element in [strings]\n * that matches this string at that position.\n */\npublic fun CharSequence.findLastAnyOf(strings: Collection<String>, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Pair<Int, String>? =\n   findAnyOf(strings, startIndex, ignoreCase, last = true)\n\n/**\n * Finds the index of the first occurrence of any of the specified [strings] in this char sequence,\n * starting from the specified [startIndex] and optionally ignoring the case.\n *\n * @param ignoreCase `true` to ignore character case when matching a string. By default `false`.\n * @return An index of the first occurrence of matched string from [strings] or -1 if none of [strings] are found.\n *\n * To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from\n * the beginning to the end of this string, and finds at each position the first element in [strings]\n * that matches this string at that position.\n */\npublic fun CharSequence.indexOfAny(strings: Collection<String>, startIndex: Int = 0, ignoreCase: Boolean = false): Int =\n   findAnyOf(strings, startIndex, ignoreCase, last = false)?.first ?: -1\n\n/**\n * Finds the index of the last occurrence of any of the specified [strings] in this char sequence,\n * starting from the specified [startIndex] and optionally ignoring the case.\n *\n * @param startIndex The index of character to start searching at. The search proceeds backward toward the beginning of the string.\n * @param ignoreCase `true` to ignore character case when matching a string. By default `false`.\n * @return An index of the last occurrence of matched string from [strings] or -1 if none of [strings] are found.\n *\n * To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from\n * the end toward the beginning of this string, and finds at each position the first element in [strings]\n * that matches this string at that position.\n */\npublic fun CharSequence.lastIndexOfAny(strings: Collection<String>, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int =\n   findAnyOf(strings, startIndex, ignoreCase, last = true)?.first ?: -1\n\n\n// indexOf\n\n/**\n * Returns the index within this string of the first occurrence of the specified character, starting from the specified [startIndex].\n *\n * @param ignoreCase `true` to ignore character case when matching a character. By default

```
`false`.\n * @return An index of the first occurrence of [char] or -1 if none is found.\n */\npublic fun CharSequence.indexOf(char: Char, startIndex: Int = 0, ignoreCase: Boolean = false): Int {\n    return if (ignoreCase || this !is String)\n        indexOfAny(charArrayOf(char), startIndex, ignoreCase)\n    else\n        nativeIndexOf(char, startIndex)\n}\n\n/**\n * Returns the index within this char sequence of the first occurrence of the specified [string],\n * starting from the specified [startIndex].\n *\n * @param ignoreCase `true` to ignore character case when matching a string. By default `false`.\n * @return An index of the first occurrence of [string] or `-1` if none is found.\n * @sample samples.text.Strings.indexOf\n */\npublic fun CharSequence.indexOf(string: String, startIndex: Int = 0, ignoreCase: Boolean = false): Int {\n    return if (ignoreCase || this !is String)\n        indexOf(string, startIndex, length, ignoreCase)\n    else\n        nativeIndexOf(string, startIndex)\n}\n\n/**\n * Returns the index within this char sequence of the last occurrence of the specified character,\n * starting from the specified [startIndex].\n *\n * @param startIndex The index of character to start searching at. The search proceeds backward toward the beginning of the string.\n * @param ignoreCase `true` to ignore character case when matching a character. By default `false`.\n * @return An index of the last occurrence of [char] or -1 if none is found.\n */\npublic fun CharSequence.lastIndexOf(char: Char, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int {\n    return if (ignoreCase || this !is String)\n        lastIndexOfAny(charArrayOf(char), startIndex, ignoreCase)\n    else\n        nativeLastIndexOf(char, startIndex)\n}\n\n/**\n * Returns the index within this char sequence of the last occurrence of the specified [string],\n * starting from the specified [startIndex].\n *\n * @param startIndex The index of character to start searching at. The search proceeds backward toward the beginning of the string.\n * @param ignoreCase `true` to ignore character case when matching a string. By default `false`.\n * @return An index of the last occurrence of [string] or -1 if none is found.\n */\npublic fun CharSequence.lastIndexOf(string: String, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int {\n    return if (ignoreCase || this !is String)\n        indexOf(string, startIndex, 0, ignoreCase, last = true)\n    else\n        nativeLastIndexOf(string, startIndex)\n}\n\n/**\n * Returns `true` if this char sequence contains the specified [other] sequence of characters as a substring.\n *\n * @param ignoreCase `true` to ignore character case when comparing strings. By default `false`.\n */\n@Suppress(\"INAPPLICABLE_OPERATOR_MODIFIER\")\npublic operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean =\n    if (other is String)\n        indexOf(other, ignoreCase = ignoreCase) >= 0\n    else\n        indexOf(other, 0, length, ignoreCase) >= 0\n\n\n\n/**\n * Returns `true` if this char sequence contains the specified character [char].\n *\n * @param ignoreCase `true` to ignore character case when comparing characters. By default `false`.\n */\n@Suppress(\"INAPPLICABLE_OPERATOR_MODIFIER\")\npublic operator fun CharSequence.contains(char: Char, ignoreCase: Boolean = false): Boolean =\n    indexOf(char, ignoreCase = ignoreCase) >= 0\n\n/**\n * Returns `true` if this char sequence contains at least one match of the specified regular expression [regex].\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun CharSequence.contains(regex: Regex): Boolean = regex.containsMatchIn(this)\n\n\n// rangesDelimitedBy\n\n\nprivate class DelimitedRangesSequence(\n    private val input: CharSequence,\n    private val startIndex: Int,\n    private val limit: Int,\n    private val getNextMatch: CharSequence.(currentIndex: Int) -> Pair<Int, Int>?\n) : Sequence<IntRange> {\n\n    override fun iterator(): Iterator<IntRange> = object : Iterator<IntRange> {\n        var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue\n        var currentStartIndex: Int = startIndex.coerceIn(0, input.length)\n        var nextSearchIndex: Int = currentStartIndex\n        var nextItem: IntRange? = null\n        var counter: Int = 0\n\n        private fun calcNext() {\n            if (nextSearchIndex < 0) {\n                nextState = 0\n                nextItem = null\n            } else {\n                if (limit > 0 && ++counter >= limit || nextSearchIndex > input.length) {\n                    nextItem = currentStartIndex..input.lastIndex\n                    nextSearchIndex = -1\n                } else {\n                    val match = input.getNextMatch(nextSearchIndex)\n                    if (match == null) {\n                        nextItem = currentStartIndex..input.lastIndex\n                        nextSearchIndex = -1\n                    } else {\n                        val (index, length) = match\n                        nextItem = currentStartIndex until index\n                        currentStartIndex = index + length\n                        nextSearchIndex = currentStartIndex + if (length == 0) 1 else 0\n                    }\n                }\n                nextState = 1\n            }\n        }\n\n        override fun next(): IntRange {\n            if (nextState == -1)\n                calcNext()\n            if (nextState == 0)\n                throw NoSuchElementException()\n            val
```

result = nextItem as IntRange\n // Clean next to avoid keeping reference on yielded instance\n nextItem = null\n nextState = -1\n return result\n }\n\n override fun hasNext(): Boolean {\n if (nextState == -1)\n calcNext()\n return nextState == 1\n }\n }\n}\n\n/**\n * Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].\n *\n * @param delimiters One or more characters to be used as delimiters.\n * @param startIndex The index to start searching delimiters from.\n * No range having its start value less than [startIndex] is returned.\n * [startIndex] is coerced to be non-negative and not greater than length of this string.\n * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`.\n * @param limit The maximum number of substrings to return. Zero by default means no limit is set.\n */\nprivate fun CharSequence.rangesDelimitedBy(delimiters: CharArray, startIndex: Int = 0, ignoreCase: Boolean = false, limit: Int = 0): Sequence<IntRange> {\n requireNonNegativeLimit(limit)\n\n return DelimitedRangesSequence(this, startIndex, limit, { currentIndex ->\n indexOfAny(delimiters, currentIndex, ignoreCase = ignoreCase).let { if (it < 0) null else it to 1 }\n })\n}\n\n\n/**\n * Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].\n *\n * @param delimiters One or more strings to be used as delimiters.\n * @param startIndex The index to start searching delimiters from.\n * No range having its start value less than [startIndex] is returned.\n * [startIndex] is coerced to be non-negative and not greater than length of this string.\n * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`.\n * @param limit The maximum number of substrings to return. Zero by default means no limit is set.\n *\n * To avoid ambiguous results when strings in [delimiters] have characters in common, this method proceeds from\n * the beginning to the end of this string, and finds at each position the first element in [delimiters]\n * that matches this string at that position.\n */\nprivate fun CharSequence.rangesDelimitedBy(delimiters: Array<out String>, startIndex: Int = 0, ignoreCase: Boolean = false, limit: Int = 0): Sequence<IntRange> {\n requireNonNegativeLimit(limit)\n val delimitersList = delimiters.asList()\n\n return DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })\n\n}\n\ninternal fun requireNonNegativeLimit(limit: Int) =\n require(limit >= 0) { \"Limit must be non-negative, but was $limit\" }\n\n\n// split\n\n/**\n * Splits this char sequence to a sequence of strings around occurrences of the specified [delimiters].\n *\n * @param delimiters One or more strings to be used as delimiters.\n * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`.\n * @param limit The maximum number of substrings to return. Zero by default means no limit is set.\n *\n * To avoid ambiguous results when strings in [delimiters] have characters in common, this method proceeds from\n * the beginning to the end of this string, and finds at each position the first element in [delimiters]\n * that matches this string at that position.\n */\npublic fun CharSequence.splitToSequence(vararg delimiters: String, ignoreCase: Boolean = false, limit: Int = 0): Sequence<String> =\n rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).map { substring(it) }\n\n/**\n * Splits this char sequence to a list of strings around occurrences of the specified [delimiters].\n *\n * @param delimiters One or more strings to be used as delimiters.\n * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`.\n * @param limit The maximum number of substrings to return. Zero by default means no limit is set.\n *\n * To avoid ambiguous results when strings in [delimiters] have characters in common, this method proceeds from\n * the beginning to the end of this string, and matches at each position the first element in [delimiters]\n * that is equal to a delimiter in this instance at that position.\n */\npublic fun CharSequence.split(vararg delimiters: String, ignoreCase: Boolean = false, limit: Int = 0): List<String> {\n if (delimiters.size == 1) {\n val delimiter = delimiters[0]\n if (!delimiter.isEmpty()) {\n return split(delimiter, ignoreCase, limit)\n }\n }\n\n return rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).asIterable().map { substring(it) }\n}\n\n/**\n * Splits this char sequence to a sequence of strings around occurrences of the specified [delimiters].\n *\n * @param delimiters One or more characters to be used as delimiters.\n * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`.\n * @param limit The maximum number of substrings to return.\n */\npublic fun CharSequence.splitToSequence(vararg delimiters: Char, ignoreCase: Boolean = false, limit: Int = 0): Sequence<String> =\n rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).map { substring(it)

```
}\n\n/**\n * Splits this char sequence to a list of strings around occurrences of the specified [delimiters].\n *\n *
@param delimiters One or more characters to be used as delimiters.\n * @param ignoreCase `true` to ignore
character case when matching a delimiter. By default `false`.\n * @param limit The maximum number of substrings
to return.\n */\npublic fun CharSequence.split(vararg delimiters: Char, ignoreCase: Boolean = false, limit: Int = 0):
List<String> {\n    if (delimiters.size == 1) {\n        return split(delimiters[0].toString(), ignoreCase, limit)\n    }\n\n
return rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).asIterable().map { substring(it)
}\n}\n\n/**\n * Splits this char sequence to a list of strings around occurrences of the specified [delimiter].\n * This
is specialized version of split which receives single non-empty delimiter and offers better performance\n *\n *
@param delimiter String used as delimiter\n * @param ignoreCase `true` to ignore character case when matching a
delimiter. By default `false`.\n * @param limit The maximum number of substrings to return.\n */\nprivate fun
CharSequence.split(delimiter: String, ignoreCase: Boolean, limit: Int): List<String> {\n
requireNonNegativeLimit(limit)\n\n    var currentOffset = 0\n    var nextIndex = indexOf(delimiter, currentOffset,
ignoreCase)\n    if (nextIndex == -1 || limit == 1) {\n        return listOf(this.toString())\n    }\n\n    val isLimited =
limit > 0\n    val result = ArrayList<String>(if (isLimited) limit.coerceAtMost(10) else 10)\n    do {\n
result.add(substring(currentOffset, nextIndex))\n        currentOffset = nextIndex + delimiter.length\n        // Do not
search for next occurrence if we're reaching limit\n        if (isLimited && result.size == limit - 1) break\n
nextIndex = indexOf(delimiter, currentOffset, ignoreCase)\n    } while (nextIndex != -1)\n\n
result.add(substring(currentOffset, length))\n    return result\n}\n\n/**\n * Splits this char sequence to a list of
strings around matches of the given regular expression.\n *\n * @param limit Non-negative value specifying the
maximum number of substrings to return.\n * Zero by default means no limit is set.\n
*/\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.split(regex: Regex, limit: Int = 0): List<String> =
regex.split(this, limit)\n\n/**\n * Splits this char sequence to a sequence of strings around matches of the given
regular expression.\n *\n * @param limit Non-negative value specifying the maximum number of substrings to
return.\n * Zero by default means no limit is set.\n * @sample samples.text.Strings.splitToSequence\n
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli
c inline fun CharSequence.splitToSequence(regex: Regex, limit: Int = 0): Sequence<String> =
regex.splitToSequence(this, limit)\n\n/**\n * Splits this char sequence to a sequence of lines delimited by any of the
following character sequences: CRLF, LF or CR.\n *\n * The lines returned do not include terminating line
separators.\n */\npublic fun CharSequence.lineSequence(): Sequence<String> = splitToSequence(\"\\r\\n\", \"\\n\",
\"\\r\")\n\n/**\n * Splits this char sequence to a list of lines delimited by any of the following character sequences:
CRLF, LF or CR.\n *\n * The lines returned do not include terminating line separators.\n */\npublic fun
CharSequence.lines(): List<String> = lineSequence().toList()\n\n/**\n * Returns `true` if the contents of this char
sequence are equal to the contents of the specified [other],\n * i.e. both char sequences contain the same number of
the same characters in the same order.\n *\n * @sample samples.text.Strings.contentEquals\n
*/\n@SinceKotlin(\"1.5\")\npublic expect infix fun CharSequence?.contentEquals(other: CharSequence?):
Boolean\n\n/**\n * Returns `true` if the contents of this char sequence are equal to the contents of the specified
[other], optionally ignoring case difference.\n *\n * @param ignoreCase `true` to ignore character case when
comparing contents.\n *\n * @sample samples.text.Strings.contentEquals\n */\n@SinceKotlin(\"1.5\")\npublic
expect fun CharSequence?.contentEquals(other: CharSequence?, ignoreCase: Boolean): Boolean\n\ninternal fun
CharSequence?.contentEqualsIgnoreCaseImpl(other: CharSequence?): Boolean {\n    if (this is String && other is
String) {\n        return this.equals(other, ignoreCase = true)\n    }\n\n    if (this === other) return true\n    if (this ==
null || other == null || this.length != other.length) return false\n\n    for (i in 0 until length) {\n        if
(!this[i].equals(other[i], ignoreCase = true)) {\n            return false\n        }\n    }\n\n    return true\n}\n\ninternal fun
CharSequence?.contentEqualsImpl(other: CharSequence?): Boolean {\n    if (this is String && other is String) {\n
return this == other\n    }\n\n    if (this === other) return true\n    if (this == null || other == null || this.length !=
other.length) return false\n\n    for (i in 0 until length) {\n        if (this[i] != other[i]) {\n            return false\n        }\n
}\n\n    return true\n}\n\n/**\n * Returns `true` if the content of this string is equal to the word \"true\", `false` if it is
equal to \"false\",\n * and throws an exception otherwise.\n *\n * There is also a lenient version of the function
```

available on nullable String, [String?.toBoolean].\n * Note that this function is case-sensitive.\n *\n * @sample samples.text.Strings.toBooleanStrict\n */\n@SinceKotlin(\"1.5\")\npublic fun String.toBooleanStrict(): Boolean = when (this) {\n \"true\" -> true\n \"false\" -> false\n else -> throw IllegalArgumentException(\"The string doesn't represent a boolean value: $this\")\n}\n\n/**\n * Returns `true` if the content of this string is equal to the word \"true\", `false` if it is equal to \"false\",\n * and `null` otherwise.\n *\n * There is also a lenient version of the function available on nullable String, [String?.toBoolean].\n * Note that this function is case-sensitive.\n *\n * @sample samples.text.Strings.toBooleanStrictOrNull\n */\n@SinceKotlin(\"1.5\")\npublic fun String.toBooleanStrictOrNull(): Boolean? = when (this) {\n \"true\" -> true\n \"false\" -> false\n else -> null\n}","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin\n\nimport kotlin.jvm.*\n\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@JvmInline\npublic value class UByteArray\n@PublishedApi\ninternal constructor(@PublishedApi internal val storage: ByteArray) : Collection<UByte> {\n\n /** Creates a new array of the specified [size], with all elements initialized to zero. */\n public constructor(size: Int) : this(ByteArray(size))\n\n /**\n * Returns the array element at the given [index]. This method can be called using the index operator.\n *\n * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n public operator fun get(index: Int): UByte = storage[index].toUByte()\n\n /**\n * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.\n *\n * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n public operator fun set(index: Int, value: UByte) {\n storage[index] = value.toByte()\n }\n\n /** Returns the number of elements in the array. */\n public override val size: Int get() = storage.size\n\n /** Creates an iterator over the elements of the array. */\n public override operator fun iterator(): kotlin.collections.Iterator<UByte> = Iterator(storage)\n\n @Suppress(\"DEPRECATION_ERROR\")\n private class Iterator(private val array: ByteArray) : UByteIterator() {\n private var index = 0\n override fun hasNext() = index < array.size\n override fun nextUByte() = if (index < array.size) array[index++].toUByte() else throw NoSuchElementException(index.toString())\n }\n\n override fun contains(element: UByte): Boolean {\n // TODO: Eliminate this check after KT-30016 gets fixed.\n // Currently JS BE does not generate special bridge method for this method.\n @Suppress(\"USELESS_CAST\")\n if ((element as Any?) !is UByte) return false\n\n return storage.contains(element.toByte())\n }\n\n override fun containsAll(elements: Collection<UByte>): Boolean {\n return (elements as Collection<*>).all { it is UByte && storage.contains(it.toByte()) }\n }\n\n override fun isEmpty(): Boolean = this.storage.size == 0\n}\n\n/**\n * Creates a new array of the specified [size], where each element is calculated by calling the specified\n * [init] function.\n *\n * The function [init] is called for each array element sequentially starting from the first one.\n * It should return the value for an array element given its index.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray(size: Int, init: (Int) -> UByte): UByteArray {\n return UByteArray(ByteArray(size) { index -> init(index).toByte()\n })\n}\n\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ubyteArrayOf(vararg elements: UByte): UByteArray = elements\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin\n\nimport kotlin.jvm.*\n\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@JvmInline\npublic value class UIntArray\n@PublishedApi\ninternal constructor(@PublishedApi internal val storage: IntArray) : Collection<UInt> {\n\n /** Creates a new array of the specified [size], with all elements initialized to zero. */\n public constructor(size: Int) : this(IntArray(size))\n\n /**\n * Returns the array element at the given [index]. This method can be called using the index operator.\n *\n * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n public

operator fun get(index: Int): UInt = storage[index].toUInt()\n\n /**\n * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.\n *\n * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n public operator fun set(index: Int, value: UInt) {\n storage[index] = value.toInt()\n }\n\n /** Returns the number of elements in the array. */\n public override val size: Int get() = storage.size\n\n /** Creates an iterator over the elements of the array. */\n public override operator fun iterator(): kotlin.collections.Iterator<UInt> = Iterator(storage)\n\n @Suppress(\"DEPRECATION_ERROR\")\n private class Iterator(private val array: IntArray) : UIntIterator() {\n private var index = 0\n override fun hasNext() = index < array.size\n override fun nextUInt() = if (index < array.size) array[index++].toUInt() else throw NoSuchElementException(index.toString())\n }\n\n override fun contains(element: UInt): Boolean {\n // TODO: Eliminate this check after KT-30016 gets fixed.\n // Currently JS BE does not generate special bridge method for this method.\n @Suppress(\"USELESS_CAST\")\n if ((element as Any?) !is UInt) return false\n\n return storage.contains(element.toInt())\n }\n\n override fun containsAll(elements: Collection<UInt>): Boolean {\n return (elements as Collection<*>).all { it is UInt && storage.contains(it.toInt()) }\n }\n\n override fun isEmpty(): Boolean = this.storage.size == 0\n}\n\n/**\n * Creates a new array of the specified [size], where each element is calculated by calling the specified\n * [init] function.\n *\n * The function [init] is called for each array element sequentially starting from the first one.\n * It should return the value for an array element given its index.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray(size: Int, init: (Int) -> UInt): UIntArray {\n return UIntArray(IntArray(size) { index -> init(index).toInt()\n})\n}\n\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun uintArrayOf(vararg elements: UInt): UIntArray = elements\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin\n\nimport kotlin.jvm.*\n\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@JvmInline\npublic value class ULongArray\n@PublishedApi\ninternal constructor(@PublishedApi internal val storage: LongArray) : Collection<ULong> {\n\n /** Creates a new array of the specified [size], with all elements initialized to zero. */\n public constructor(size: Int) : this(LongArray(size))\n\n /**\n * Returns the array element at the given [index]. This method can be called using the index operator.\n *\n * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n public operator fun get(index: Int): ULong = storage[index].toULong()\n\n /**\n * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.\n *\n * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n public operator fun set(index: Int, value: ULong) {\n storage[index] = value.toLong()\n }\n\n /** Returns the number of elements in the array. */\n public override val size: Int get() = storage.size\n\n /** Creates an iterator over the elements of the array. */\n public override operator fun iterator(): kotlin.collections.Iterator<ULong> = Iterator(storage)\n\n @Suppress(\"DEPRECATION_ERROR\")\n private class Iterator(private val array: LongArray) : ULongIterator() {\n private var index = 0\n override fun hasNext() = index < array.size\n override fun nextULong() = if (index < array.size) array[index++].toULong() else throw NoSuchElementException(index.toString())\n }\n\n override fun contains(element: ULong): Boolean {\n // TODO: Eliminate this check after KT-30016 gets fixed.\n // Currently JS BE does not generate special bridge method for this method.\n @Suppress(\"USELESS_CAST\")\n if ((element as Any?) !is ULong) return false\n\n return storage.contains(element.toLong())\n }\n\n override fun containsAll(elements: Collection<ULong>): Boolean {\n return (elements as Collection<*>).all { it is ULong && storage.contains(it.toLong()) }\n }\n\n override fun isEmpty(): Boolean = this.storage.size == 0\n}\n\n/**\n * Creates a new array of the specified [size], where each element is calculated by calling the specified\n * [init] function.\n *\n * The function [init] is called for each array element sequentially starting from the first one.\n * It

should return the value for an array element given its index.\n
/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {\n   return ULongArray(LongArray(size) { index -> init(index).toLong()\n})\n}\n\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements\n","/\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin\n\nimport kotlin.jvm.*\n\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@JvmInline\npublic value class UShortArray\n@PublishedApi\ninternal constructor(@PublishedApi internal val storage: ShortArray) : Collection<UShort> {\n\n   /** Creates a new array of the specified [size], with all elements initialized to zero. */\n   public constructor(size: Int) : this(ShortArray(size))\n\n   /**\n   * Returns the array element at the given [index]. This method can be called using the index operator.\n   *\n   * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n   * where the behavior is unspecified.\n   */\n   public operator fun get(index: Int): UShort = storage[index].toUShort()\n\n   /**\n   * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.\n   *\n   * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n   * where the behavior is unspecified.\n   */\n   public operator fun set(index: Int, value: UShort) {\n   storage[index] = value.toShort()\n   }\n\n   /** Returns the number of elements in the array. */\n   public override val size: Int get() = storage.size\n\n   /** Creates an iterator over the elements of the array. */\n   public override operator fun iterator(): kotlin.collections.Iterator<UShort> = Iterator(storage)\n\n   @Suppress(\"DEPRECATION_ERROR\")\n   private class Iterator(private val array: ShortArray) : UShortIterator() {\n   private var index = 0\n   override fun hasNext() = index < array.size\n   override fun nextUShort() = if (index < array.size) array[index++].toUShort() else throw NoSuchElementException(index.toString())\n   }\n\n   override fun contains(element: UShort): Boolean {\n   // TODO: Eliminate this check after KT-30016 gets fixed.\n   // Currently JS BE does not generate special bridge method for this method.\n   @Suppress(\"USELESS_CAST\")\n   if ((element as Any?) !is UShort) return false\n\n   return storage.contains(element.toShort())\n   }\n\n   override fun containsAll(elements: Collection<UShort>): Boolean {\n   return (elements as Collection<*>).all { it is UShort && storage.contains(it.toShort()) }\n   }\n\n   override fun isEmpty(): Boolean = this.storage.size == 0\n}\n\n/**\n * Creates a new array of the specified [size], where each element is calculated by calling the specified\n * [init] function.\n *\n * The function [init] is called for each array element sequentially starting from the first one.\n * It should return the value for an array element given its index.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {\n   return UShortArray(ShortArray(size) { index -> init(index).toShort()\n})\n}\n\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"UArraysKt\")\n@file:kotlin.jvm.JvmPackageName(\"kotlin.collections.unsigned\")\n\npackage kotlin.collections\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\nimport kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntArray.component1(): UInt {\n   return get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n

* where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongArray.component1(): ULong {\n   return get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UByteArray.component1(): UByte {\n   return get(0)\n}\n\n/**\n * Returns 1st *element* from the array.\n * \n * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UShortArray.component1(): UShort {\n   return get(0)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntArray.component2(): UInt {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongArray.component2(): ULong {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UByteArray.component2(): UByte {\n   return get(1)\n}\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UShortArray.component2(): UShort {\n   return get(1)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntArray.component3(): UInt {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongArray.component3(): ULong {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UByteArray.component3(): UByte {\n   return get(2)\n}\n\n/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UShortArray.component3(): UShort {\n   return get(2)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntArray.component4(): UInt {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior

is unspecified.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongArray.component4(): ULong {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UByteArray.component4(): UByte {\n   return get(3)\n}\n\n/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UShortArray.component4(): UShort {\n   return get(3)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntArray.component5(): UInt {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongArray.component5(): ULong {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UByteArray.component5(): UByte {\n   return get(4)\n}\n\n/**\n * Returns 5th *element* from the array.\n * \n * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UShortArray.component5(): UShort {\n   return get(4)\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic expect fun UIntArray.elementAt(index: Int): UInt\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic expect fun ULongArray.elementAt(index: Int): ULong\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic expect fun UByteArray.elementAt(index: Int): UByte\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic expect fun UShortArray.elementAt(index: Int): UShort\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the

result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {\n   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.elementAtOrNull(index: Int): UInt? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.elementAtOrNull(index: Int): ULong? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.elementAtOrNull(index: Int): UByte? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAtOrNull\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.elementAtOrNull(index: Int): UShort? {\n   return this.getOrNull(index)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.find(predicate: (UInt) -> Boolean): UInt? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.find(predicate: (ULong) -> Boolean): ULong? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.find(predicate: (UByte) -> Boolean): UByte? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.find(predicate: (UShort) -> Boolean): UShort? {\n   return firstOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.findLast(predicate: (UInt) -> Boolean): UInt? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n

*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.findLast(predicate: (ULong) -> Boolean): ULong? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.findLast(predicate: (UByte) -> Boolean): UByte? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.find\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.findLast(predicate: (UShort) -> Boolean): UShort? {\n   return lastOrNull(predicate)\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.first(): UInt {\n   return storage.first().toUInt()\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.first(): ULong {\n   return storage.first().toULong()\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.first(): UByte {\n   return storage.first().toUByte()\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.first(): UShort {\n   return storage.first().toUShort()\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.first(predicate: (UInt) -> Boolean): UInt {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.first(predicate: (ULong) -> Boolean): ULong {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.first(predicate: (UByte) -> Boolean): UByte {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.first(predicate: (UShort) -> Boolean): UShort {\n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.firstOrNull(): UInt? {\n   return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.firstOrNull(): ULong? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.firstOrNull(): UByte? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n

*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.firstOrNull(): UShort? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.firstOrNull(predicate: (UInt) -> Boolean): UInt? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.firstOrNull(predicate: (ULong) -> Boolean): ULong? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.firstOrNull(predicate: (UByte) -> Boolean): UByte? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.firstOrNull(predicate: (UShort) -> Boolean): UShort? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.getOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {\n return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.getOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {\n return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.getOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {\n return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.getOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {\n return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.getOrNull(index: Int): UInt? {\n return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.getOrNull(index: Int): ULong? {\n return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.getOrNull(index: Int): UByte? {\n return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.getOrNull\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.getOrNull(index: Int): UShort? {\n return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns first index of

[element], or -1 if the array does not contain element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.indexOf(element: UInt): Int {\n   return storage.indexOf(element.toInt())\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.indexOf(element: ULong): Int {\n   return storage.indexOf(element.toLong())\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.indexOf(element: UByte): Int {\n   return storage.indexOf(element.toByte())\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.indexOf(element: UShort): Int {\n   return storage.indexOf(element.toShort())\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.indexOfFirst(predicate: (UInt) -> Boolean): Int {\n   return storage.indexOfFirst { predicate(it.toUInt()) }\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.indexOfFirst(predicate: (ULong) -> Boolean): Int {\n   return storage.indexOfFirst { predicate(it.toULong()) }\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int {\n   return storage.indexOfFirst { predicate(it.toUByte()) }\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.indexOfFirst(predicate: (UShort) -> Boolean): Int {\n   return storage.indexOfFirst { predicate(it.toUShort()) }\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.indexOfLast(predicate: (UInt) -> Boolean): Int {\n   return storage.indexOfLast { predicate(it.toUInt()) }\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.indexOfLast(predicate: (ULong) -> Boolean): Int {\n   return storage.indexOfLast { predicate(it.toULong()) }\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.indexOfLast(predicate: (UByte) -> Boolean): Int {\n   return storage.indexOfLast { predicate(it.toUByte()) }\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.indexOfLast(predicate: (UShort) -> Boolean): Int {\n   return storage.indexOfLast { predicate(it.toUShort()) }\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.last(): UInt {\n   return storage.last().toUInt()\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.last(): ULong {\n   return storage.last().toULong()\n}\n\n/**\n * Returns the last element.\n * \n *

@throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.last(): UByte {\n   return storage.last().toUByte()\n}\n\n/**\n * Returns the last element.\n * \n * @throws NoSuchElementException if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.last(): UShort {\n   return storage.last().toUShort()\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.last(predicate: (UInt) -> Boolean): UInt {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.last(predicate: (ULong) -> Boolean): ULong {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.last(predicate: (UByte) -> Boolean): UByte {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * \n * @throws NoSuchElementException if no such element is found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.last(predicate: (UShort) -> Boolean): UShort {\n   for (index in this.indices.reversed()) {\n   val element = this[index]\n   if (predicate(element)) return element\n   }\n   throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.lastIndexOf(element: UInt): Int {\n   return storage.lastIndexOf(element.toInt())\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.lastIndexOf(element: ULong): Int {\n   return storage.lastIndexOf(element.toLong())\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.lastIndexOf(element: UByte): Int {\n   return storage.lastIndexOf(element.toByte())\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.lastIndexOf(element: UShort): Int {\n   return storage.lastIndexOf(element.toShort())\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.lastOrNull(): UInt? {\n   return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n

*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.lastOrNull(): ULong? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.lastOrNull(): UByte? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.lastOrNull(): UShort? {\n return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.lastOrNull(predicate: (UInt) -> Boolean): UInt? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.lastOrNull(predicate: (ULong) -> Boolean): ULong? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.lastOrNull(predicate: (UByte) -> Boolean): UByte? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n * \n * @sample samples.collections.Collections.Elements.last\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.lastOrNull(predicate: (UShort) -> Boolean): UShort? {\n for (index in this.indices.reversed()) {\n val element = this[index]\n if (predicate(element)) return element\n }\n return null\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.random(): UInt {\n return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.random(): ULong {\n return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.random(): UByte {\n return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.random(): UShort {\n return random(Random)\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.random(random: Random): UInt {\n if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.random(random: Random): ULong {\n if (isEmpty())\n throw NoSuchElementException(\"Array is empty.\")\n return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */

*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.random(random: Random): UByte {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.random(random: Random): UShort {\n   if (isEmpty())\n   throw NoSuchElementException(\"Array is empty.\")\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.randomOrNull(): UInt? {\n   return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.randomOrNull(): ULong? {\n   return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.randomOrNull(): UByte? {\n   return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.randomOrNull(): UShort? {\n   return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun UIntArray.randomOrNull(random: Random): UInt? {\n   if (isEmpty())\n   return null\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun ULongArray.randomOrNull(random: Random): ULong? {\n   if (isEmpty())\n   return null\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun UByteArray.randomOrNull(random: Random): UByte? {\n   if (isEmpty())\n   return null\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun UShortArray.randomOrNull(random: Random): UShort? {\n   if (isEmpty())\n   return null\n   return get(random.nextInt(size))\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.single(): UInt {\n   return storage.single().toUInt()\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.single(): ULong {\n   return storage.single().toULong()\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.single(): UByte {\n   return storage.single().toUByte()\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.single(): UShort {\n   return storage.single().toUShort()\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n

*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.single(predicate: (UInt) -> Boolean): UInt {\n var single: UInt? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as UInt\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.single(predicate: (ULong) -> Boolean): ULong {\n var single: ULong? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as ULong\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.single(predicate: (UByte) -> Boolean): UByte {\n var single: UByte? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as UByte\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.single(predicate: (UShort) -> Boolean): UShort {\n var single: UShort? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) throw IllegalArgumentException(\"Array contains more than one matching element.\")\n single = element\n found = true\n }\n }\n if (!found) throw NoSuchElementException(\"Array contains no element matching the predicate.\")\n @Suppress(\"UNCHECKED_CAST\")\n return single as UShort\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.singleOrNull(): UInt? {\n return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.singleOrNull(): ULong? {\n return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.singleOrNull(): UByte? {\n return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or `null` if the array is empty or has more than one element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.singleOrNull(): UShort? {\n return if (size == 1) this[0] else null\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.singleOrNull(predicate: (UInt) -> Boolean): UInt? {\n var single: UInt? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) return null\n single = element\n found = true\n }\n }\n if (!found) return null\n return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.singleOrNull(predicate: (ULong) -> Boolean): ULong? {\n var single: ULong? = null\n var found = false\n for (element in this) {\n if (predicate(element)) {\n if (found) return null\n single = element\n found = true\n }\n }\n if (!found) return null\n return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was

found.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.singleOrNull(predicate: (UByte) -> Boolean): UByte? {\n   var single: UByte? = null\n   var found = false\n   for (element in this) {\n   if (predicate(element)) {\n   if (found) return null\n   single = element\n   found = true\n   }\n   }\n   if (!found) return null\n   return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.singleOrNull(predicate: (UShort) -> Boolean): UShort? {\n   var single: UShort? = null\n   var found = false\n   for (element in this) {\n   if (predicate(element)) {\n   if (found) return null\n   single = element\n   found = true\n   }\n   }\n   if (!found) return null\n   return single\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.drop(n: Int): List<UInt> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.drop(n: Int): List<ULong> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.drop(n: Int): List<UByte> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.drop(n: Int): List<UShort> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return takeLast((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.dropLast(n: Int): List<UInt> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.dropLast(n: Int): List<ULong> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.dropLast(n: Int): List<UByte> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size - n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.dropLast(n: Int): List<UShort> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   return take((size -

n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.dropLastWhile(predicate: (UInt) -> Boolean): List<UInt> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.dropLastWhile(predicate: (ULong) -> Boolean): List<ULong> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.dropLastWhile(predicate: (UByte) -> Boolean): List<UByte> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.dropLastWhile(predicate: (UShort) -> Boolean): List<UShort> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.dropWhile(predicate: (UInt) -> Boolean): List<UInt> {\n var yielding = false\n val list = ArrayList<UInt>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.dropWhile(predicate: (ULong) -> Boolean): List<ULong> {\n var yielding = false\n val list = ArrayList<ULong>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.dropWhile(predicate: (UByte) -> Boolean): List<UByte> {\n var yielding = false\n val list = ArrayList<UByte>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.dropWhile(predicate: (UShort) -> Boolean): List<UShort> {\n var yielding = false\n val list = ArrayList<UShort>()\n for (item in this)\n if (yielding)\n list.add(item)\n else if (!predicate(item)) {\n list.add(item)\n yielding = true\n }\n return list\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.filter(predicate: (UInt) -> Boolean): List<UInt> {\n return filterTo(ArrayList<UInt>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample

samples.collections.Collections.Filtering.filter\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.filter(predicate: (ULong) -> Boolean): List<ULong> {\n   return filterTo(ArrayList<ULong>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.filter(predicate: (UByte) -> Boolean): List<UByte> {\n   return filterTo(ArrayList<UByte>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {\n   return filterTo(ArrayList<UShort>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.filterIndexed(predicate: (index: Int, UInt) -> Boolean): List<UInt> {\n   return filterIndexedTo(ArrayList<UInt>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.filterIndexed(predicate: (index: Int, ULong) -> Boolean): List<ULong> {\n   return filterIndexedTo(ArrayList<ULong>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.filterIndexed(predicate: (index: Int, UByte) -> Boolean): List<UByte> {\n   return filterIndexedTo(ArrayList<UByte>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexed\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.filterIndexed(predicate: (index: Int, UShort) -> Boolean): List<UShort> {\n   return filterIndexedTo(ArrayList<UShort>(), predicate)\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in UInt>> UIntArray.filterIndexedTo(destination: C, predicate: (index: Int, UInt) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in ULong>> ULongArray.filterIndexedTo(destination: C, predicate: (index: Int, ULong) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given

[predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in UByte>> UByteArray.filterIndexedTo(destination: C, predicate: (index: Int, UByte) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in UShort>> UShortArray.filterIndexedTo(destination: C, predicate: (index: Int, UShort) -> Boolean): C {\n   forEachIndexed { index, element ->\n   if (predicate(index, element)) destination.add(element)\n   }\n   return destination\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.filterNot(predicate: (UInt) -> Boolean): List<UInt> {\n   return filterNotTo(ArrayList<UInt>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.filterNot(predicate: (ULong) -> Boolean): List<ULong> {\n   return filterNotTo(ArrayList<ULong>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.filterNot(predicate: (UByte) -> Boolean): List<UByte> {\n   return filterNotTo(ArrayList<UByte>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.filterNot(predicate: (UShort) -> Boolean): List<UShort> {\n   return filterNotTo(ArrayList<UShort>(), predicate)\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in UInt>> UIntArray.filterNotTo(destination: C, predicate: (UInt) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in ULong>> ULongArray.filterNotTo(destination: C, predicate: (ULong) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in UByte>> UByteArray.filterNotTo(destination: C, predicate: (UByte) -> Boolean): C {\n   for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in UShort>> UShortArray.filterNotTo(destination: C, predicate: (UShort) -> Boolean): C {\n

for (element in this) if (!predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in UInt>> UIntArray.filterTo(destination: C, predicate: (UInt) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in ULong>> ULongArray.filterTo(destination: C, predicate: (ULong) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in UByte>> UByteArray.filterTo(destination: C, predicate: (UByte) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection<in UShort>> UShortArray.filterTo(destination: C, predicate: (UShort) -> Boolean): C {\n   for (element in this) if (predicate(element)) destination.add(element)\n   return destination\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.slice(indices: IntRange): List<UInt> {\n   if (indices.isEmpty()) return listOf()\n   return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.slice(indices: IntRange): List<ULong> {\n   if (indices.isEmpty()) return listOf()\n   return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.slice(indices: IntRange): List<UByte> {\n   if (indices.isEmpty()) return listOf()\n   return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.slice(indices: IntRange): List<UShort> {\n   if (indices.isEmpty()) return listOf()\n   return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.slice(indices: Iterable<Int>): List<UInt> {\n   val size = indices.collectionSizeOrDefault(10)\n   if (size == 0) return emptyList()\n   val list = ArrayList<UInt>(size)\n   for (index in indices) {\n   list.add(get(index))\n   }\n   return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.slice(indices: Iterable<Int>): List<ULong> {\n   val size = indices.collectionSizeOrDefault(10)\n   if (size == 0) return emptyList()\n   val list = ArrayList<ULong>(size)\n   for (index in indices) {\n   list.add(get(index))\n   }\n   return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.slice(indices: Iterable<Int>): List<UByte> {\n   val size = indices.collectionSizeOrDefault(10)\n   if (size == 0) return emptyList()\n   val list = ArrayList<UByte>(size)\n   for (index in indices) {\n   list.add(get(index))\n   }\n   return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.slice(indices: Iterable<Int>): List<UShort> {\n   val size = indices.collectionSizeOrDefault(10)\n   if (size == 0) return emptyList()\n   val list =

ArrayList<UShort>(size)\n   for (index in indices) {\n   list.add(get(index))\n   }\n   return list\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sliceArray(indices: Collection<Int>): UIntArray {\n   return UIntArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sliceArray(indices: Collection<Int>): ULongArray {\n   return ULongArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sliceArray(indices: Collection<Int>): UByteArray {\n   return UByteArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sliceArray(indices: Collection<Int>): UShortArray {\n   return UShortArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sliceArray(indices: IntRange): UIntArray {\n   return UIntArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sliceArray(indices: IntRange): ULongArray {\n   return ULongArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sliceArray(indices: IntRange): UByteArray {\n   return UByteArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sliceArray(indices: IntRange): UShortArray {\n   return UShortArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.take(n: Int): List<UInt> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   if (n == 0) return emptyList()\n   if (n >= size) return toList()\n   if (n == 1) return listOf(this[0])\n   var count = 0\n   val list = ArrayList<UInt>(n)\n   for (item in this) {\n   list.add(item)\n   if (++count == n)\n   break\n   }\n   return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.take(n: Int): List<ULong> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   if (n == 0) return emptyList()\n   if (n >= size) return toList()\n   if (n == 1) return listOf(this[0])\n   var count = 0\n   val list = ArrayList<ULong>(n)\n   for (item in this) {\n   list.add(item)\n   if (++count == n)\n   break\n   }\n   return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.take(n: Int): List<UByte> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   if (n == 0) return emptyList()\n   if (n >= size) return toList()\n   if (n == 1) return listOf(this[0])\n   var count = 0\n   val list = ArrayList<UByte>(n)\n   for (item in this) {\n   list.add(item)\n   if (++count == n)\n   break\n   }\n   return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.take(n: Int): List<UShort> {\n   require(n >= 0) { \"Requested element count $n is less than zero.\" }\n   if (n == 0) return emptyList()\n   if (n >= size) return toList()\n   if (n == 1) return listOf(this[0])\n   var count = 0\n   val list = ArrayList<UShort>(n)\n   for

(item in this) {\n list.add(item)\n if (++count == n)\n break\n }\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.takeLast(n: Int): List<UInt> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<UInt>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.takeLast(n: Int): List<ULong> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<ULong>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.takeLast(n: Int): List<UByte> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<UByte>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.takeLast(n: Int): List<UShort> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<UShort>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.takeLastWhile(predicate: (UInt) -> Boolean): List<UInt> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return drop(index + 1)\n }\n }\n return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.takeLastWhile(predicate: (ULong) -> Boolean): List<ULong> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return drop(index + 1)\n }\n }\n return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.takeLastWhile(predicate: (UByte) -> Boolean): List<UByte> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return drop(index + 1)\n }\n }\n return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.takeLastWhile(predicate: (UShort) -> Boolean): List<UShort> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n return drop(index + 1)\n }\n }\n return toList()\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

UIntArray.takeWhile(predicate: (UInt) -> Boolean): List<UInt> {\n val list = ArrayList<UInt>()\n for (item in this) {\n if (!predicate(item))\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.takeWhile(predicate: (ULong) -> Boolean): List<ULong> {\n val list = ArrayList<ULong>()\n for (item in this) {\n if (!predicate(item))\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.takeWhile(predicate: (UByte) -> Boolean): List<UByte> {\n val list = ArrayList<UByte>()\n for (item in this) {\n if (!predicate(item))\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.takeWhile(predicate: (UShort) -> Boolean): List<UShort> {\n val list = ArrayList<UShort>()\n for (item in this) {\n if (!predicate(item))\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reverse(): Unit {\n storage.reverse()\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reverse(): Unit {\n storage.reverse()\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reverse(): Unit {\n storage.reverse()\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reverse(): Unit {\n storage.reverse()\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n storage.reverse(fromIndex, toIndex)\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n storage.reverse(fromIndex, toIndex)\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n storage.reverse(fromIndex, toIndex)\n}\n\n/**\n * Reverses elements of the array in the specified range in-place.\n * \n * @param fromIndex the start of the range (inclusive) to reverse.\n * @param toIndex the end of the range (exclusive) to reverse.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

UShortArray.reverse(fromIndex: Int, toIndex: Int): Unit {\n   storage.reverse(fromIndex, toIndex)\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.reversed(): List<UInt> {\n   if (isEmpty()) return emptyList()\n   val list = toMutableList()\n   list.reverse()\n   return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.reversed(): List<ULong> {\n   if (isEmpty()) return emptyList()\n   val list = toMutableList()\n   list.reverse()\n   return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.reversed(): List<UByte> {\n   if (isEmpty()) return emptyList()\n   val list = toMutableList()\n   list.reverse()\n   return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.reversed(): List<UShort> {\n   if (isEmpty()) return emptyList()\n   val list = toMutableList()\n   list.reverse()\n   return list\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reversedArray(): UIntArray {\n   return UIntArray(storage.reversedArray())\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reversedArray(): ULongArray {\n   return ULongArray(storage.reversedArray())\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reversedArray(): UByteArray {\n   return UByteArray(storage.reversedArray())\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reversedArray(): UShortArray {\n   return UShortArray(storage.reversedArray())\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.shuffle(): Unit {\n   shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.shuffle(): Unit {\n   shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.shuffle(): Unit {\n   shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.shuffle(): Unit {\n   shuffle(Random)\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.shuffle(random: Random): Unit {\n   for (i in lastIndex downTo 1) {\n   val j = random.nextInt(i + 1)\n   val copy = this[i]\n   this[i] = this[j]\n   this[j] = copy\n   }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.shuffle(random: Random): Unit {\n   for (i in lastIndex downTo 1) {\n   val j = random.nextInt(i + 1)\n   val copy = this[i]\n   this[i] = this[j]\n   this[j] = copy\n   }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See: https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.shuffle(random: Random): Unit {\n   for (i in lastIndex downTo 1) {\n   val j = random.nextInt(i + 1)\n   val copy = this[i]\n   this[i] = this[j]\n   this[j] = copy\n   }\n}\n\n/**\n * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.\n * \n * See:

https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle#The_modern_algorithm\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.shuffle(random: Random): Unit {\n for (i in lastIndex downTo 1) {\n val j = random.nextInt(i + 1)\n val copy = this[i]\n this[i] = this[j]\n this[j] = copy\n }\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n }\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sorted(): List<UInt> {\n return copyOf().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sorted(): List<ULong> {\n return copyOf().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sorted(): List<UByte> {\n return copyOf().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sorted(): List<UShort> {\n return copyOf().apply { sort() }.asList()\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sortedArray(): UIntArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sortedArray(): ULongArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sortedArray(): UByteArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sortedArray(): UShortArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sortedArrayDescending(): UIntArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sortedArrayDescending(): ULongArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sortedArrayDescending(): UByteArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns an array with all elements of this array sorted descending according to their natural sort order.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sortedArrayDescending(): UShortArray {\n if (isEmpty()) return this\n return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It

means that equal elements preserve their order relative to each other after sorting.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sortedDescending(): List<UInt> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sortedDescending(): List<ULong> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sortedDescending(): List<UByte> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sortedDescending(): List<UShort> {\n   return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns an array of type [ByteArray], which is a view of this array where each element is a signed reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.asByteArray(): ByteArray {\n   return storage\n}\n\n/**\n * Returns an array of type [IntArray], which is a view of this array where each element is a signed reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.asIntArray(): IntArray {\n   return storage\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic expect fun UIntArray.asList(): List<UInt>\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic expect fun ULongArray.asList(): List<ULong>\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic expect fun UByteArray.asList(): List<UByte>\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic expect fun UShortArray.asList(): List<UShort>\n\n/**\n * Returns an array of type [LongArray], which is a view of this array where each element is a signed reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.asLongArray(): LongArray {\n   return storage\n}\n\n/**\n * Returns an array of type [ShortArray], which is a view of this array where each element is a signed reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.asShortArray(): ShortArray {\n   return storage\n}\n\n/**\n * Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.asUByteArray(): UByteArray {\n   return UByteArray(this)\n}\n\n/**\n * Returns an array of type [UIntArray], which is a view of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.asUIntArray(): UIntArray {\n   return UIntArray(this)\n}\n\n/**\n * Returns an array of type [ULongArray], which is a view of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.asULongArray(): ULongArray {\n   return ULongArray(this)\n}\n\n/**\n * Returns an array of type [UShortArray], which is a view of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

ShortArray.asUShortArray(): UShortArray {\n   return UShortArray(this)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\npublic infix fun UIntArray.contentEquals(other: UIntArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\npublic infix fun ULongArray.contentEquals(other: ULongArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\npublic infix fun UByteArray.contentEquals(other: UByteArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\npublic infix fun UShortArray.contentEquals(other: UShortArray): Boolean {\n   return this.contentEquals(other)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic infix fun UIntArray?.contentEquals(other: UIntArray?): Boolean {\n   return this?.storage.contentEquals(other?.storage)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic infix fun ULongArray?.contentEquals(other: ULongArray?): Boolean {\n   return this?.storage.contentEquals(other?.storage)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic infix fun UByteArray?.contentEquals(other: UByteArray?): Boolean {\n   return this?.storage.contentEquals(other?.storage)\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic infix fun UShortArray?.contentEquals(other: UShortArray?): Boolean {\n   return this?.storage.contentEquals(other?.storage)\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.contentHashCode(): Int {\n   return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.contentHashCode(): Int {\n   return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.contentHashCode(): Int {\n   return this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation

warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince =
\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.contentHashCode(): Int {\n   return
this.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray?.contentHashCode(): Int {\n
return this?.storage.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it
is [List].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun
ULongArray?.contentHashCode(): Int {\n   return this?.storage.contentHashCode()\n}\n\n/**\n * Returns a hash
code based on the contents of this array as if it is [List].\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray?.contentHashCode(): Int {\n
return this?.storage.contentHashCode()\n}\n\n/**\n * Returns a hash code based on the contents of this array as if it
is [List].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun
UShortArray?.contentHashCode(): Int {\n   return this?.storage.contentHashCode()\n}\n\n/**\n * Returns a string
representation of the contents of the specified array as if it is [List].\n * \n * @sample
samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to
avoid deprecation warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince =
\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.contentToString(): String {\n   return
this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is
[List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use
Kotlin compiler 1.4 to avoid deprecation
warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince =
\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.contentToString(): String {\n   return
this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is
[List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use
Kotlin compiler 1.4 to avoid deprecation
warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince =
\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.contentToString(): String {\n   return
this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is
[List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@Deprecated(\"Use
Kotlin compiler 1.4 to avoid deprecation
warning.\")\n@SinceKotlin(\"1.3\")\n@DeprecatedSinceKotlin(hiddenSince =
\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.contentToString(): String {\n   return
this.contentToString()\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is
[List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray?.contentToString(): String {\n
return this?.joinToString(\", \", \"[\", \"]\") ?: \"null\"\n}\n\n/**\n * Returns a string representation of the contents of
the specified array as if it is [List].\n * \n * @sample
samples.collections.Arrays.ContentOperations.contentToString\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray?.contentToString(): String {\n
return this?.joinToString(\", \", \"[\", \"]\") ?: \"null\"\n}\n\n/**\n * Returns a string representation of the contents of
the specified array as if it is [List].\n * \n * @sample
samples.collections.Arrays.ContentOperations.contentToString\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray?.contentToString(): String {\n
return this?.joinToString(\", \", \"[\", \"]\") ?: \"null\"\n}\n\n/**\n * Returns a string representation of the contents of
the specified array as if it is [List].\n * \n * @sample
samples.collections.Arrays.ContentOperations.contentToString\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray?.contentToString(): String {\n
return this?.joinToString(\", \", \"[\", \"]\") ?: \"null\"\n}\n\n/**\n * Copies this array or its subrange into the
[destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even

specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.copyInto(destination: UIntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UIntArray {\n   storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.copyInto(destination: ULongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ULongArray {\n   storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.copyInto(destination: UByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UByteArray {\n   storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.copyInto(destination: UShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UShortArray {\n   storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n   return destination\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample

samples.collections.Arrays.CopyOfOperations.copyOf\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.copyOf(): UIntArray {\n   return UIntArray(storage.copyOf())\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.copyOf(): ULongArray {\n   return ULongArray(storage.copyOf())\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.copyOf(): UByteArray {\n   return UByteArray(storage.copyOf())\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.copyOf(): UShortArray {\n   return UShortArray(storage.copyOf())\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.copyOf(newSize: Int): UIntArray {\n   return UIntArray(storage.copyOf(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.copyOf(newSize: Int): ULongArray {\n   return ULongArray(storage.copyOf(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.copyOf(newSize: Int): UByteArray {\n   return UByteArray(storage.copyOf(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.copyOf(newSize: Int): UShortArray {\n   return UShortArray(storage.copyOf(newSize))\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.copyOfRange(fromIndex: Int, toIndex: Int): UIntArray {\n   return UIntArray(storage.copyOfRange(fromIndex, toIndex))\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

ULongArray.copyOfRange(fromIndex: Int, toIndex: Int): ULongArray {\n   return ULongArray(storage.copyOfRange(fromIndex, toIndex))\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {\n   return UByteArray(storage.copyOfRange(fromIndex, toIndex))\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive) to copy.\n * @param toIndex the end of the range (exclusive) to copy.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.copyOfRange(fromIndex: Int, toIndex: Int): UShortArray {\n   return UShortArray(storage.copyOfRange(fromIndex, toIndex))\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.fill(element: UInt, fromIndex: Int = 0, toIndex: Int = size): Unit {\n   storage.fill(element.toInt(), fromIndex, toIndex)\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.fill(element: ULong, fromIndex: Int = 0, toIndex: Int = size): Unit {\n   storage.fill(element.toLong(), fromIndex, toIndex)\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.fill(element: UByte, fromIndex: Int = 0, toIndex: Int = size): Unit {\n   storage.fill(element.toByte(), fromIndex, toIndex)\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive) to fill, 0 by default.\n * @param toIndex the end of the range (exclusive) to fill, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.fill(element: UShort, fromIndex: Int = 0, toIndex: Int = size): Unit {\n   storage.fill(element.toShort(), fromIndex, toIndex)\n}\n\n/**\n * Returns the range of valid indices for the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic inline val UIntArray.indices: IntRange\n   get() = storage.indices\n\n/**\n * Returns the range of valid indices for the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic inline val ULongArray.indices: IntRange\n   get() = storage.indices\n\n/**\n * Returns the range of valid indices for the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic inline val UByteArray.indices: IntRange\n   get() = storage.indices\n\n/**\n * Returns the range of valid indices for the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic inline val UShortArray.indices: IntRange\n

get() = storage.indices\n\n/**\n * Returns the last valid index for the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic inline val UIntArray.lastIndex: Int\n   get() = storage.lastIndex\n\n/**\n * Returns the last valid index for the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic inline val ULongArray.lastIndex: Int\n   get() = storage.lastIndex\n\n/**\n * Returns the last valid index for the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic inline val UByteArray.lastIndex: Int\n   get() = storage.lastIndex\n\n/**\n * Returns the last valid index for the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic inline val UShortArray.lastIndex: Int\n   get() = storage.lastIndex\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntArray.plus(element: UInt): UIntArray {\n   return UIntArray(storage + element.toInt())\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongArray.plus(element: ULong): ULongArray {\n   return ULongArray(storage + element.toLong())\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UByteArray.plus(element: UByte): UByteArray {\n   return UByteArray(storage + element.toByte())\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UShortArray.plus(element: UShort): UShortArray {\n   return UShortArray(storage + element.toShort())\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic operator fun UIntArray.plus(elements: Collection<UInt>): UIntArray {\n   var index = size\n   val result = storage.copyOf(size + elements.size)\n   for (element in elements) result[index++] = element.toInt()\n   return UIntArray(result)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic operator fun ULongArray.plus(elements: Collection<ULong>): ULongArray {\n   var index = size\n   val result = storage.copyOf(size + elements.size)\n   for (element in elements) result[index++] = element.toLong()\n   return ULongArray(result)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic operator fun UByteArray.plus(elements: Collection<UByte>): UByteArray {\n   var index = size\n   val result = storage.copyOf(size + elements.size)\n   for (element in elements) result[index++] = element.toByte()\n   return UByteArray(result)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic operator fun UShortArray.plus(elements: Collection<UShort>): UShortArray {\n   var index = size\n   val result = storage.copyOf(size + elements.size)\n   for (element in elements) result[index++] = element.toShort()\n   return UShortArray(result)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntArray.plus(elements: UIntArray): UIntArray {\n   return UIntArray(storage + elements.storage)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongArray.plus(elements: ULongArray): ULongArray {\n   return ULongArray(storage + elements.storage)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UByteArray.plus(elements: UByteArray): UByteArray {\n   return UByteArray(storage + elements.storage)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements

of the given [elements] array.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UShortArray.plus(elements: UShortArray): UShortArray {\n   return UShortArray(storage + elements.storage)\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sort(): Unit {\n   if (size > 1) sortArray(this, 0, size)\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sort(): Unit {\n   if (size > 1) sortArray(this, 0, size)\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sort(): Unit {\n   if (size > 1) sortArray(this, 0, size)\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sort(): Unit {\n   if (size > 1) sortArray(this, 0, size)\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   sortArray(this, fromIndex, toIndex)\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   sortArray(this, fromIndex, toIndex)\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   sortArray(this, fromIndex, toIndex)\n}\n\n/**\n * Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {\n   AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n   sortArray(this, fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n   sort(fromIndex, toIndex)\n   reverse(fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of

the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n   sort(fromIndex, toIndex)\n   reverse(fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n   sort(fromIndex, toIndex)\n   reverse(fromIndex, toIndex)\n}\n\n/**\n * Sorts elements of the array in the specified range in-place.\n * The elements are sorted descending according to their natural sort order.\n * \n * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n   sort(fromIndex, toIndex)\n   reverse(fromIndex, toIndex)\n}\n\n/**\n * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.toByteArray(): ByteArray {\n   return storage.copyOf()\n}\n\n/**\n * Returns an array of type [IntArray], which is a copy of this array where each element is a signed reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.toIntArray(): IntArray {\n   return storage.copyOf()\n}\n\n/**\n * Returns an array of type [LongArray], which is a copy of this array where each element is a signed reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.toLongArray(): LongArray {\n   return storage.copyOf()\n}\n\n/**\n * Returns an array of type [ShortArray], which is a copy of this array where each element is a signed reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.toShortArray(): ShortArray {\n   return storage.copyOf()\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.toTypedArray(): Array<UInt> {\n   return Array(size) { index -> this[index] }\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.toTypedArray(): Array<ULong> {\n   return Array(size) { index -> this[index] }\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.toTypedArray(): Array<UByte> {\n   return Array(size) { index -> this[index] }\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.toTypedArray(): Array<UShort> {\n   return Array(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of UByte containing all of the elements of this generic array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Array<out UByte>.toUByteArray(): UByteArray {\n   return UByteArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of type [UByteArray], which is a copy of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n

*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.toUByteArray(): UByteArray {\n   return UByteArray(this.copyOf())\n}\n\n/**\n * Returns an array of UInt containing all of the elements of this generic array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Array<out UInt>.toUIntArray(): UIntArray {\n   return UIntArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of type [UIntArray], which is a copy of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.toUIntArray(): UIntArray {\n   return UIntArray(this.copyOf())\n}\n\n/**\n * Returns an array of ULong containing all of the elements of this generic array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Array<out ULong>.toULongArray(): ULongArray {\n   return ULongArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of type [ULongArray], which is a copy of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.toULongArray(): ULongArray {\n   return ULongArray(this.copyOf())\n}\n\n/**\n * Returns an array of UShort containing all of the elements of this generic array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Array<out UShort>.toUShortArray(): UShortArray {\n   return UShortArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of type [UShortArray], which is a copy of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.toUShortArray(): UShortArray {\n   return UShortArray(this.copyOf())\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> UIntArray.associateWith(valueSelector: (UInt) -> V): Map<UInt, V> {\n   val result = LinkedHashMap<UInt, V>(mapCapacity(size).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> ULongArray.associateWith(valueSelector: (ULong) -> V): Map<ULong, V> {\n   val result = LinkedHashMap<ULong, V>(mapCapacity(size).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> UByteArray.associateWith(valueSelector: (UByte) -> V): Map<UByte, V> {\n   val result = LinkedHashMap<UByte, V>(mapCapacity(size).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V>

UShortArray.associateWith(valueSelector: (UShort) -> V): Map<UShort, V> {\n   val result = LinkedHashMap<UShort, V>(mapCapacity(size).coerceAtLeast(16))\n   return associateWithTo(result, valueSelector)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in UInt, in V>> UIntArray.associateWithTo(destination: M, valueSelector: (UInt) -> V): M {\n   for (element in this) {\n   destination.put(element, valueSelector(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in ULong, in V>> ULongArray.associateWithTo(destination: M, valueSelector: (ULong) -> V): M {\n   for (element in this) {\n   destination.put(element, valueSelector(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in UByte, in V>> UByteArray.associateWithTo(destination: M, valueSelector: (UByte) -> V): M {\n   for (element in this) {\n   destination.put(element, valueSelector(element))\n   }\n   return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * @sample samples.collections.Collections.Transformations.associateWithTo\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V, M : MutableMap<in UShort, in V>> UShortArray.associateWithTo(destination: M, valueSelector: (UShort) -> V): M {\n   for (element in this) {\n   destination.put(element, valueSelector(element))\n   }\n   return destination\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.flatMap(transform: (UInt) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.flatMap(transform: (ULong) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.flatMap(transform: (UByte) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMap\n

*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.flatMap(transform: (UShort) -> Iterable<R>): List<R> {\n   return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.flatMapIndexed(transform: (index: Int, UInt) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.flatMapIndexed(transform: (index: Int, ULong) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.flatMapIndexed(transform: (index: Int, UByte) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array.\n * \n * @sample samples.collections.Collections.Transformations.flatMapIndexed\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.flatMapIndexed(transform: (index: Int, UShort) -> Iterable<R>): List<R> {\n   return flatMapIndexedTo(ArrayList<R>(), transform)\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UIntArray.flatMapIndexedTo(destination: C, transform: (index: Int, UInt) -> Iterable<R>): C {\n   var index = 0\n   for (element in this) {\n   val list = transform(index++, element)\n   destination.addAll(list)\n   }\n   return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> ULongArray.flatMapIndexedTo(destination: C, transform: (index: Int, ULong) -> Iterable<R>): C {\n   var index = 0\n   for (element in this) {\n   val list = transform(index++, element)\n   destination.addAll(list)\n   }\n   return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UByteArray.flatMapIndexedTo(destination: C, transform: (index: Int, UByte) -> Iterable<R>): C {\n   var index = 0\n   for (element in this) {\n   val list = transform(index++, element)\n   destination.addAll(list)\n   }\n   return destination\n}\n\n/**\n * Appends all elements yielded from results of

[transform] function being invoked on each element\n * and its index in the original array, to the given [destination].\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UShortArray.flatMapIndexedTo(destination: C, transform: (index: Int, UShort) -> Iterable<R>): C {\n   var index = 0\n   for (element in this) {\n   val list = transform(index++, element)\n   destination.addAll(list)\n   }\n   return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UIntArray.flatMapTo(destination: C, transform: (UInt) -> Iterable<R>): C {\n   for (element in this) {\n   val list = transform(element)\n   destination.addAll(list)\n   }\n   return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> ULongArray.flatMapTo(destination: C, transform: (ULong) -> Iterable<R>): C {\n   for (element in this) {\n   val list = transform(element)\n   destination.addAll(list)\n   }\n   return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UByteArray.flatMapTo(destination: C, transform: (UByte) -> Iterable<R>): C {\n   for (element in this) {\n   val list = transform(element)\n   destination.addAll(list)\n   }\n   return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UShortArray.flatMapTo(destination: C, transform: (UShort) -> Iterable<R>): C {\n   for (element in this) {\n   val list = transform(element)\n   destination.addAll(list)\n   }\n   return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K> UIntArray.groupBy(keySelector: (UInt) -> K): Map<K, List<UInt>> {\n   return groupByTo(LinkedHashMap<K, MutableList<UInt>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K> ULongArray.groupBy(keySelector: (ULong) -> K): Map<K, List<ULong>> {\n   return groupByTo(LinkedHashMap<K, MutableList<ULong>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K> UByteArray.groupBy(keySelector: (UByte) -> K): Map<K, List<UByte>> {\n   return groupByTo(LinkedHashMap<K, MutableList<UByte>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the

entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K> UShortArray.groupBy(keySelector: (UShort) -> K): Map<K, List<UShort>> {\n return groupByTo(LinkedHashMap<K, MutableList<UShort>>(), keySelector)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> UIntArray.groupBy(keySelector: (UInt) -> K, valueTransform: (UInt) -> V): Map<K, List<V>> {\n return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> ULongArray.groupBy(keySelector: (ULong) -> K, valueTransform: (ULong) -> V): Map<K, List<V>> {\n return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> UByteArray.groupBy(keySelector: (UByte) -> K, valueTransform: (UByte) -> V): Map<K, List<V>> {\n return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> UShortArray.groupBy(keySelector: (UShort) -> K, valueTransform: (UShort) -> V): Map<K, List<V>> {\n return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, M : MutableMap<in K, MutableList<UInt>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<UInt>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, M : MutableMap<in K, MutableList<ULong>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K):

M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<ULong>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, M : MutableMap<in K, MutableList<UByte>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<UByte>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, M : MutableMap<in K, MutableList<UShort>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<UShort>() }\n list.add(element)\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K, valueTransform: (UInt) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K, valueTransform: (ULong) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K, valueTransform: (UByte) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K, valueTransform: (UShort) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return

destination\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.map(transform: (UInt) -> R): List<R> {\n   return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.map(transform: (ULong) -> R): List<R> {\n   return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.map(transform: (UByte) -> R): List<R> {\n   return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.map(transform: (UShort) -> R): List<R> {\n   return mapTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.mapIndexed(transform: (index: Int, UInt) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.mapIndexed(transform: (index: Int, ULong) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.mapIndexed(transform: (index: Int, UByte) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.mapIndexed(transform: (index: Int, UShort) -> R): List<R> {\n   return mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UIntArray.mapIndexedTo(destination: C, transform: (index: Int, UInt) -> R): C {\n   var index = 0\n   for (item in this)\n   destination.add(transform(index++, item))\n   return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> ULongArray.mapIndexedTo(destination: C, transform: (index: Int, ULong) -> R): C {\n

```
 var index = 0\n    for (item in this)\n        destination.add(transform(index++, item))\n    return
destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n
* and appends the results to the given [destination].\n * @param [transform] function that takes the index of an
element and the element itself\n * and returns the result of the transform applied to the element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UByteArray.mapIndexedTo(destination: C, transform: (index: Int, UByte) -> R): C {\n
var index = 0\n    for (item in this)\n        destination.add(transform(index++, item))\n    return
destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n
* and appends the results to the given [destination].\n * @param [transform] function that takes the index of an
element and the element itself\n * and returns the result of the transform applied to the element.\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UShortArray.mapIndexedTo(destination: C, transform: (index: Int, UShort) -> R): C {\n
 var index = 0\n    for (item in this)\n        destination.add(transform(index++, item))\n    return
destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and
appends the results to the given [destination].\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UIntArray.mapTo(destination: C, transform: (UInt) -> R): C {\n    for (item in this)\n
destination.add(transform(item))\n    return destination\n}\n\n/**\n * Applies the given [transform] function to each
element of the original array\n * and appends the results to the given [destination].\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> ULongArray.mapTo(destination: C, transform: (ULong) -> R): C {\n    for (item in
this)\n        destination.add(transform(item))\n    return destination\n}\n\n/**\n * Applies the given [transform]
function to each element of the original array\n * and appends the results to the given [destination].\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UByteArray.mapTo(destination: C, transform: (UByte) -> R): C {\n    for (item in this)\n
    destination.add(transform(item))\n    return destination\n}\n\n/**\n * Applies the given [transform] function to
each element of the original array\n * and appends the results to the given [destination].\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UShortArray.mapTo(destination: C, transform: (UShort) -> R): C {\n    for (item in
this)\n        destination.add(transform(item))\n    return destination\n}\n\n/**\n * Returns a lazy [Iterable] that wraps
each element of the original array\n * into an [IndexedValue] containing the index of that element and the element
itself.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.withIndex():
Iterable<IndexedValue<UInt>> {\n    return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable]
that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the
element itself.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.withIndex():
Iterable<IndexedValue<ULong>> {\n    return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable]
that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the
element itself.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.withIndex():
Iterable<IndexedValue<UByte>> {\n    return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable]
that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the
element itself.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.withIndex():
Iterable<IndexedValue<UShort>> {\n    return IndexingIterable { iterator() }\n}\n\n/**\n * Returns `true` if all
elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.all(predicate: (UInt) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return
false\n    return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.all\n
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

ULongArray.all(predicate: (ULong) -> Boolean): Boolean {\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.all(predicate: (UByte) -> Boolean): Boolean {\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.all(predicate: (UShort) -> Boolean): Boolean {\n for (element in this) if (!predicate(element)) return false\n return true\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.any(): Boolean {\n return storage.any()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.any(): Boolean {\n return storage.any()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.any(): Boolean {\n return storage.any()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.any(): Boolean {\n return storage.any()\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.any(predicate: (UInt) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.any(predicate: (ULong) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.any(predicate: (UByte) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.any(predicate: (UShort) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return false\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.count(predicate: (UInt) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.count(predicate: (ULong) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.count(predicate: (UByte) -> Boolean): Int {\n var count = 0\n for (element in this) if (predicate(element)) ++count\n return count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic

inline fun UShortArray.count(predicate: (UShort) -> Boolean): Int {\n   var count = 0\n   for (element in this) if (predicate(element)) ++count\n   return count\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.fold(initial: R, operation: (acc: R, UInt) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.fold(initial: R, operation: (acc: R, ULong) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.fold(initial: R, operation: (acc: R, UByte) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.fold(initial: R, operation: (acc: R, UShort) -> R): R {\n   var accumulator = initial\n   for (element in this) accumulator = operation(accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): R {\n   var index = 0\n   var accumulator = initial\n   for (element in this) accumulator = operation(index++, accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to\n right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): R {\n   var index = 0\n   var accumulator = initial\n   for (element in this) accumulator = operation(index++, accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to\n right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {\n   var index = 0\n   var accumulator = initial\n   for (element in this) accumulator = operation(index++, accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to

right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UShort) -> R): R {\n   var index = 0\n   var accumulator = initial\n   for (element in this) accumulator = operation(index++, accumulator, element)\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.foldRight(initial: R, operation: (UInt, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.foldRight(initial: R, operation: (ULong, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.foldRight(initial: R, operation: (UByte, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.foldRight(initial: R, operation: (UShort, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.foldRightIndexed(initial: R, operation: (index: Int, UInt, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.foldRightIndexed(initial: R, operation: (index: Int, ULong, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(index, get(index),

accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.foldRightIndexed(initial: R, operation: (index: Int, UShort, acc: R) -> R): R {\n   var index = lastIndex\n   var accumulator = initial\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Performs the given [action] on each element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.forEach(action: (UInt) -> Unit): Unit {\n   for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.forEach(action: (ULong) -> Unit): Unit {\n   for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.forEach(action: (UByte) -> Unit): Unit {\n   for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.forEach(action: (UShort) -> Unit): Unit {\n   for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.forEachIndexed(action: (index: Int, UInt) -> Unit): Unit {\n   var index = 0\n   for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.forEachIndexed(action: (index: Int, ULong) -> Unit): Unit {\n   var index = 0\n   for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.forEachIndexed(action: (index: Int, UByte) -> Unit): Unit {\n   var index = 0\n   for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.forEachIndexed(action: (index: Int, UShort) -> Unit): Unit {\n   var index = 0\n   for (item in this) action(index++, item)\n}\n\n@Deprecated(\"Use maxOrNull instead.\",

ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.max(): UInt? {\n   return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.max(): ULong? {\n   return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.max(): UByte? {\n   return maxOrNull()\n}\n\n@Deprecated(\"Use maxOrNull instead.\", ReplaceWith(\"this.maxOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.max(): UShort? {\n   return maxOrNull()\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UIntArray.maxBy(selector: (UInt) -> R): UInt? {\n   return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ULongArray.maxBy(selector: (ULong) -> R): ULong? {\n   return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UByteArray.maxBy(selector: (UByte) -> R): UByte? {\n   return maxByOrNull(selector)\n}\n\n@Deprecated(\"Use maxByOrNull instead.\", ReplaceWith(\"this.maxByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.maxBy(selector: (UShort) -> R): UShort? {\n   return maxByOrNull(selector)\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UIntArray.maxByOrNull(selector: (UInt) -> R): UInt? {\n   if (isEmpty()) return null\n   var maxElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return maxElem\n   var maxValue = selector(maxElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (maxValue < v) {\n   maxElem = e\n   maxValue = v\n   }\n   }\n   return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ULongArray.maxByOrNull(selector: (ULong) -> R): ULong? {\n   if (isEmpty()) return null\n   var maxElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return maxElem\n   var maxValue = selector(maxElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (maxValue < v) {\n   maxElem = e\n   maxValue = v\n   }\n   }\n   return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R :

Comparable<R>> UByteArray.maxByOrNull(selector: (UByte) -> R): UByte? {\n   if (isEmpty()) return null\n   var maxElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return maxElem\n   var maxValue = selector(maxElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (maxValue < v) {\n   maxElem = e\n   maxValue = v\n   }\n   }\n   return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxByOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.maxByOrNull(selector: (UShort) -> R): UShort? {\n   if (isEmpty()) return null\n   var maxElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return maxElem\n   var maxValue = selector(maxElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (maxValue < v) {\n   maxElem = e\n   maxValue = v\n   }\n   }\n   return maxElem\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.maxOf(selector: (UInt) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.maxOf(selector: (ULong) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.maxOf(selector: (UByte) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.maxOf(selector: (UShort) -> Double): Double {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.maxOf(selector: (UInt) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced

by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.maxOf(selector: (ULong) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.maxOf(selector: (UByte) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.maxOf(selector: (UShort) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UIntArray.maxOf(selector: (UInt) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ULongArray.maxOf(selector: (ULong) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UByteArray.maxOf(selector: (UByte) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.maxOf(selector: (UShort) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v =

selector(this[i])\n if (maxValue < v) {\n maxValue = v\n }\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.maxOfOrNull(selector: (UInt) -> Double): Double? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.maxOfOrNull(selector: (ULong) -> Double): Double? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.maxOfOrNull(selector: (UByte) -> Double): Double? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.maxOfOrNull(selector: (UShort) -> Double): Double? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.maxOfOrNull(selector: (UInt) -> Float): Float? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.maxOfOrNull(selector: (ULong) -> Float): Float? {\n if (isEmpty()) return null\n var maxValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n maxValue = maxOf(maxValue, v)\n }\n return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

UByteArray.maxOfOrNull(selector: (UByte) -> Float): Float? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.maxOfOrNull(selector: (UShort) -> Float): Float? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   maxValue = maxOf(maxValue, v)\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UIntArray.maxOfOrNull(selector: (UInt) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ULongArray.maxOfOrNull(selector: (ULong) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UByteArray.maxOfOrNull(selector: (UByte) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.maxOfOrNull(selector: (UShort) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (maxValue < v) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.maxOfWith(comparator: Comparator<in R>, selector: (UInt) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.maxOfWith(comparator: Comparator<in R>, selector: (ULong) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v =

selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.maxOfWith(comparator: Comparator<in R>, selector: (UByte) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.maxOfWith(comparator: Comparator<in R>, selector: (UShort) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (UInt) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (ULong) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (UByte) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (UShort) -> R): R? {\n   if (isEmpty()) return null\n   var maxValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(maxValue, v) < 0) {\n   maxValue = v\n   }\n   }\n   return maxValue\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.maxOrNull(): UInt? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (max < e) max

= e\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.maxOrNull(): ULong? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (max < e) max = e\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.maxOrNull(): UByte? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (max < e) max = e\n   }\n   return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.maxOrNull(): UShort? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (max < e) max = e\n   }\n   return max\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.maxWith(comparator: Comparator<in UInt>): UInt? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.maxWith(comparator: Comparator<in ULong>): ULong? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.maxWith(comparator: Comparator<in UByte>): UByte? {\n   return maxWithOrNull(comparator)\n}\n\n@Deprecated(\"Use maxWithOrNull instead.\", ReplaceWith(\"this.maxWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.maxWith(comparator: Comparator<in UShort>): UShort? {\n   return maxWithOrNull(comparator)\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.maxWithOrNull(comparator: Comparator<in UInt>): UInt? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(max, e) < 0) max = e\n   }\n   return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.maxWithOrNull(comparator: Comparator<in ULong>): ULong? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(max, e) < 0) max = e\n   }\n   return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.maxWithOrNull(comparator: Comparator<in UByte>): UByte? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(max, e) < 0) max = e\n   }\n   return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.maxWithOrNull(comparator: Comparator<in UShort>): UShort? {\n   if (isEmpty()) return null\n   var max = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(max, e) < 0) max = e\n   }\n   return max\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.min(): UInt? {\n   return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\", ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.min():

ULong? {\n   return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\",
ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\",
hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.min():
UByte? {\n   return minOrNull()\n}\n\n@Deprecated(\"Use minOrNull instead.\",
ReplaceWith(\"this.minOrNull()\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\",
hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.min():
UShort? {\n   return minOrNull()\n}\n\n@Deprecated(\"Use minByOrNull instead.\",
ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince =
\"1.5\", hiddenSince =
\"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
<R : Comparable<R>> UIntArray.minBy(selector: (UInt) -> R): UInt? {\n   return
minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\",
ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince =
\"1.5\", hiddenSince =
\"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
<R : Comparable<R>> ULongArray.minBy(selector: (ULong) -> R): ULong? {\n   return
minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\",
ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince =
\"1.5\", hiddenSince =
\"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
<R : Comparable<R>> UByteArray.minBy(selector: (UByte) -> R): UByte? {\n   return
minByOrNull(selector)\n}\n\n@Deprecated(\"Use minByOrNull instead.\",
ReplaceWith(\"this.minByOrNull(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince =
\"1.5\", hiddenSince =
\"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
<R : Comparable<R>> UShortArray.minBy(selector: (UShort) -> R): UShort? {\n   return
minByOrNull(selector)\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or
`null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.minByOrNull\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R :
Comparable<R>> UIntArray.minByOrNull(selector: (UInt) -> R): UInt? {\n   if (isEmpty()) return null\n   var
minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue =
selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v)
{\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first
element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.minByOrNull\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R :
Comparable<R>> ULongArray.minByOrNull(selector: (ULong) -> R): ULong? {\n   if (isEmpty()) return null\n
var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue =
selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v)
{\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first
element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.minByOrNull\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R :
Comparable<R>> UByteArray.minByOrNull(selector: (UByte) -> R): UByte? {\n   if (isEmpty()) return null\n
var minElem = this[0]\n   val lastIndex = this.lastIndex\n   if (lastIndex == 0) return minElem\n   var minValue =
selector(minElem)\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   val v = selector(e)\n   if (minValue > v)
{\n   minElem = e\n   minValue = v\n   }\n   }\n   return minElem\n}\n\n/**\n * Returns the first
element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample

samples.collections.Collections.Aggregates.minByOrNull\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.minByOrNull(selector: (UShort) -> R): UShort? {\n if (isEmpty()) return null\n var minElem = this[0]\n val lastIndex = this.lastIndex\n if (lastIndex == 0) return minElem\n var minValue = selector(minElem)\n for (i in 1..lastIndex) {\n val e = this[i]\n val v = selector(e)\n if (minValue > v) {\n minElem = e\n minValue = v\n }\n }\n return minElem\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.minOf(selector: (UInt) -> Double): Double {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.minOf(selector: (ULong) -> Double): Double {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.minOf(selector: (UByte) -> Double): Double {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.minOf(selector: (UShort) -> Double): Double {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.minOf(selector: (UInt) -> Float): Float {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue = minOf(minValue, v)\n }\n return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.minOf(selector: (ULong) -> Float): Float {\n if (isEmpty()) throw NoSuchElementException()\n var minValue = selector(this[0])\n for (i in 1..lastIndex) {\n val v = selector(this[i])\n minValue =

minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.minOf(selector: (UByte) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.minOf(selector: (UShort) -> Float): Float {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UIntArray.minOf(selector: (UInt) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ULongArray.minOf(selector: (ULong) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UByteArray.minOf(selector: (UByte) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.minOf(selector: (UShort) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

UIntArray.minOfOrNull(selector: (UInt) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.minOfOrNull(selector: (ULong) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.minOfOrNull(selector: (UByte) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.minOfOrNull(selector: (UShort) -> Double): Double? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.minOfOrNull(selector: (UInt) -> Float): Float? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.minOfOrNull(selector: (ULong) -> Float): Float? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.minOfOrNull(selector: (UByte) -> Float): Float? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.minOfOrNull(selector: (UShort) -> Float): Float? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   minValue = minOf(minValue, v)\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UIntArray.minOfOrNull(selector: (UInt) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ULongArray.minOfOrNull(selector: (ULong) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UByteArray.minOfOrNull(selector: (UByte) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.minOfOrNull(selector: (UShort) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (minValue > v) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n *\n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.minOfWith(comparator: Comparator<in R>, selector: (UInt) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n *\n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.minOfWith(comparator: Comparator<in R>, selector: (ULong) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n *\n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>

UByteArray.minOfWith(comparator: Comparator<in R>, selector: (UByte) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.minOfWith(comparator: Comparator<in R>, selector: (UShort) -> R): R {\n   if (isEmpty()) throw NoSuchElementException()\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (UInt) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (ULong) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (UByte) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest value according to the provided [comparator]\n * among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (UShort) -> R): R? {\n   if (isEmpty()) return null\n   var minValue = selector(this[0])\n   for (i in 1..lastIndex) {\n   val v = selector(this[i])\n   if (comparator.compare(minValue, v) > 0) {\n   minValue = v\n   }\n   }\n   return minValue\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.minOrNull(): UInt? {\n   if (isEmpty()) return null\n   var min = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (min > e) min = e\n   }\n   return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.minOrNull(): ULong? {\n   if (isEmpty()) return null\n   var min = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (min > e) min = e\n   }\n   return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.minOrNull(): UByte? {\n   if (isEmpty()) return null\n   var min = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (min > e) min =

e\n   }\n   return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.minOrNull(): UShort? {\n   if (isEmpty()) return null\n   var min = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (min > e) min = e\n   }\n   return min\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.minWith(comparator: Comparator<in UInt>): UInt? {\n   return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.minWith(comparator: Comparator<in ULong>): ULong? {\n   return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.minWith(comparator: Comparator<in UByte>): UByte? {\n   return minWithOrNull(comparator)\n}\n\n@Deprecated(\"Use minWithOrNull instead.\", ReplaceWith(\"this.minWithOrNull(comparator)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.5\", hiddenSince = \"1.6\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.minWith(comparator: Comparator<in UShort>): UShort? {\n   return minWithOrNull(comparator)\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.minWithOrNull(comparator: Comparator<in UInt>): UInt? {\n   if (isEmpty()) return null\n   var min = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(min, e) > 0) min = e\n   }\n   return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.minWithOrNull(comparator: Comparator<in ULong>): ULong? {\n   if (isEmpty()) return null\n   var min = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(min, e) > 0) min = e\n   }\n   return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.minWithOrNull(comparator: Comparator<in UByte>): UByte? {\n   if (isEmpty()) return null\n   var min = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(min, e) > 0) min = e\n   }\n   return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.minWithOrNull(comparator: Comparator<in UShort>): UShort? {\n   if (isEmpty()) return null\n   var min = this[0]\n   for (i in 1..lastIndex) {\n   val e = this[i]\n   if (comparator.compare(min, e) > 0) min = e\n   }\n   return min\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample samples.collections.Collections.Aggregates.none\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.none(): Boolean {\n   return isEmpty()\n}\n\n/**\n * Returns `true` if no elements match the given

[predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.none(predicate: (UInt) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.none(predicate: (ULong) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.none(predicate: (UByte) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.none(predicate: (UShort) -> Boolean): Boolean {\n   for (element in this) if (predicate(element)) return false\n   return true\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.onEach(action: (UInt) -> Unit): UIntArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.onEach(action: (ULong) -> Unit): ULongArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.onEach(action: (UByte) -> Unit): UByteArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element and returns the array itself afterwards.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.onEach(action: (UShort) -> Unit): UShortArray {\n   return apply { for (element in this) action(element) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.onEachIndexed(action: (index: Int, UInt) -> Unit): UIntArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.onEachIndexed(action: (index: Int, ULong) -> Unit): ULongArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.onEachIndexed(action: (index: Int, UByte) -> Unit): UByteArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.onEachIndexed(action: (index: Int, UShort) -> Unit): UShortArray {\n   return apply { forEachIndexed(action) }\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation]

from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reduce(operation: (acc: UInt, UInt) -> UInt): UInt {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reduce(operation: (acc: ULong, ULong) -> ULong): ULong {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reduce(operation: (acc: UShort, UShort) -> UShort): UShort {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt {\n   if (isEmpty())\n   throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(index, accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull]

instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reduceIndexed(operation: (index: Int, acc: UShort, UShort) -> UShort): UShort {\n if (isEmpty())\n throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reduceIndexedOrNull(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reduceIndexedOrNull(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong? {\n if (isEmpty())\n return null\n var accumulator = this[0]\n for (index in 1..lastIndex) {\n accumulator = operation(index, accumulator, this[index])\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n

*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reduceIndexedOrNull(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(index, accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reduceIndexedOrNull(operation: (index: Int, acc: UShort, UShort) -> UShort): UShort? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(index, accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reduceOrNull(operation: (acc: UInt, UInt) -> UInt): UInt? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reduceOrNull(operation: (acc: ULong, ULong) -> ULong): ULong? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reduceOrNull(operation: (acc: UByte, UByte) -> UByte): UByte? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reduceOrNull(operation: (acc: UShort, UShort) -> UShort): UShort? {\n   if (isEmpty())\n   return null\n   var accumulator = this[0]\n   for (index in 1..lastIndex) {\n   accumulator = operation(accumulator, this[index])\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator

value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reduceRight(operation: (UInt, acc: UInt) -> UInt): UInt {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reduceRight(operation: (ULong, acc: ULong) -> ULong): ULong {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reduceRight(operation: (UByte, acc: UByte) -> UByte): UByte {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reduceRight(operation: (UShort, acc: UShort) -> UShort): UShort {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(get(index--), accumulator)\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reduceRightIndexed(operation: (index: Int, UInt, acc: UInt) -> UInt): UInt {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use

[reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reduceRightIndexed(operation: (index: Int, ULong, acc: ULong) -> ULong): ULong {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reduceRightIndexed(operation: (index: Int, UByte, acc: UByte) -> UByte): UByte {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reduceRightIndexed(operation: (index: Int, UShort, acc: UShort) -> UShort): UShort {\n   var index = lastIndex\n   if (index < 0) throw UnsupportedOperationException(\"Empty array can't be reduced.\")\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reduceRightIndexedOrNull(operation: (index: Int, UInt, acc: UInt) -> UInt): UInt? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reduceRightIndexedOrNull(operation: (index: Int, ULong, acc: ULong) -> ULong): ULong? {\n   var index = lastIndex\n   if (index < 0) return null\n   var accumulator = get(index--)\n   while (index >= 0) {\n   accumulator = operation(index, get(index), accumulator)\n   --index\n   }\n   return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with

its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reduceRightIndexedOrNull(operation: (index: Int, UByte, acc: UByte) -> UByte): UByte? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reduceRightIndexedOrNull(operation: (index: Int, UShort, acc: UShort) -> UShort): UShort? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reduceRightOrNull(operation: (UInt, acc: UInt) -> UInt): UInt? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reduceRightOrNull(operation: (ULong, acc: ULong) -> ULong): ULong? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reduceRightOrNull(operation: (UByte, acc: UByte) -> UByte): UByte? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with the last element and applying [operation] from right to left\n * to each element and current accumulator value.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reduceRightOrNull(operation: (UShort, acc: UShort) -> UShort): UShort? {\n var index = lastIndex\n if (index < 0) return null\n var accumulator = get(index--)\n

while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.runningFold(initial: R, operation: (acc: R, UInt) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (element in this) {\n accumulator = operation(accumulator, element)\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.runningFold(initial: R, operation: (acc: R, ULong) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (element in this) {\n accumulator = operation(accumulator, element)\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.runningFold(initial: R, operation: (acc: R, UByte) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (element in this) {\n accumulator = operation(accumulator, element)\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.runningFold(initial: R, operation: (acc: R, UShort) -> R): List<R> {\n if (isEmpty()) return listOf(initial)\n val result = ArrayList<R>(size + 1).apply { add(initial) }\n var accumulator = initial\n for (element in this) {\n accumulator = operation(accumulator, element)\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>

UIntArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (index in indices) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (index in indices) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (index in indices) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, UShort) -> R): List<R> {\n   if (isEmpty()) return listOf(initial)\n   val result = ArrayList<R>(size + 1).apply { add(initial) }\n   var accumulator = initial\n   for (index in indices) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.runningReduce(operation: (acc: UInt, UInt) -> UInt): List<UInt> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<UInt>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample

samples.collections.Collections.Aggregates.runningReduce\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.runningReduce(operation: (acc: ULong, ULong) -> ULong): List<ULong> {\n if (isEmpty()) return emptyList()\n var accumulator = this[0]\n val result = ArrayList<ULong>(size).apply { add(accumulator) }\n for (index in 1 until size) {\n accumulator = operation(accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.runningReduce(operation: (acc: UByte, UByte) -> UByte): List<UByte> {\n if (isEmpty()) return emptyList()\n var accumulator = this[0]\n val result = ArrayList<UByte>(size).apply { add(accumulator) }\n for (index in 1 until size) {\n accumulator = operation(accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.runningReduce(operation: (acc: UShort, UShort) -> UShort): List<UShort> {\n if (isEmpty()) return emptyList()\n var accumulator = this[0]\n val result = ArrayList<UShort>(size).apply { add(accumulator) }\n for (index in 1 until size) {\n accumulator = operation(accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.runningReduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): List<UInt> {\n if (isEmpty()) return emptyList()\n var accumulator = this[0]\n val result = ArrayList<UInt>(size).apply { add(accumulator) }\n for (index in 1 until size) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.runningReduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): List<ULong> {\n if (isEmpty()) return emptyList()\n var accumulator = this[0]\n val result = ArrayList<ULong>(size).apply { add(accumulator) }\n for (index in 1 until size) {\n accumulator = operation(index, accumulator, this[index])\n result.add(accumulator)\n }\n return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and

current accumulator value that starts with the first element of this array.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.runningReduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): List<UByte> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<UByte>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.runningReduceIndexed(operation: (index: Int, acc: UShort, UShort) -> UShort): List<UShort> {\n   if (isEmpty()) return emptyList()\n   var accumulator = this[0]\n   val result = ArrayList<UShort>(size).apply { add(accumulator) }\n   for (index in 1 until size) {\n   accumulator = operation(index, accumulator, this[index])\n   result.add(accumulator)\n   }\n   return result\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.scan(initial: R, operation: (acc: R, UInt) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.scan(initial: R, operation: (acc: R, ULong) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.scan(initial: R, operation: (acc: R, UByte) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current

accumulator value and an element, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.scan(initial: R, operation: (acc: R, UShort) -> R): List<R> {\n   return runningFold(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): List<R> {\n   return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.scanIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): List<R> {\n   return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): List<R> {\n   return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UShort) -> R): List<R> {\n   return runningFoldIndexed(initial, operation)\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumBy(selector: (UInt) -> UInt): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

ULongArray.sumBy(selector: (ULong) -> UInt): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumBy(selector: (UByte) -> UInt): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumBy(selector: (UShort) -> UInt): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumByDouble(selector: (UInt) -> Double): Double {\n   var sum: Double = 0.0\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumByDouble(selector: (ULong) -> Double): Double {\n   var sum: Double = 0.0\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumByDouble(selector: (UByte) -> Double): Double {\n   var sum: Double = 0.0\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@Deprecated(\"Use sumOf instead.\", ReplaceWith(\"this.sumOf(selector)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumByDouble(selector: (UShort) -> Double): Double {\n   var sum: Double = 0.0\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumOf(selector: (UInt) -> Double): Double {\n   var sum: Double = 0.toDouble()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumOf(selector: (ULong) -> Double): Double {\n   var sum: Double = 0.toDouble()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumOf(selector:

(UByte) -> Double): Double {\n   var sum: Double = 0.toDouble()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfDouble\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> Double): Double {\n   var sum: Double = 0.toDouble()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumOf(selector: (UInt) -> Int): Int {\n   var sum: Int = 0.toInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumOf(selector: (ULong) -> Int): Int {\n   var sum: Int = 0.toInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumOf(selector: (UByte) -> Int): Int {\n   var sum: Int = 0.toInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfInt\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> Int): Int {\n   var sum: Int = 0.toInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumOf(selector: (UInt) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumOf(selector: (ULong) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumOf(selector:

(UByte) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.4\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfLong\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> Long): Long {\n   var sum: Long = 0.toLong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumOf(selector: (UInt) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumOf(selector: (ULong) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumOf(selector: (UByte) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> UInt): UInt {\n   var sum: UInt = 0.toUInt()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumOf(selector: (UInt) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumOf(selector: (ULong) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.Inli

neOnly\npublic inline fun UByteArray.sumOf(selector: (UByte) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\n@SinceKotlin(\"1.5\")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolutionByLambdaReturnType\n@Suppress(\"INAPPLICABLE_JVM_NAME\")\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@ExperimentalUnsignedTypes\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> ULong): ULong {\n   var sum: ULong = 0.toULong()\n   for (element in this) {\n   sum += selector(element)\n   }\n   return sum\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UIntArray.zip(other: Array<out R>): List<Pair<UInt, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun <R> ULongArray.zip(other: Array<out R>): List<Pair<ULong, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UByteArray.zip(other: Array<out R>): List<Pair<UByte, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UShortArray.zip(other: Array<out R>): List<Pair<UShort, R>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> UIntArray.zip(other: Array<out R>, transform: (a: UInt, b: R) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> ULongArray.zip(other: Array<out R>, transform: (a: ULong, b: R) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> UByteArray.zip(other: Array<out R>, transform: (a: UByte, b: R) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n

*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> UShortArray.zip(other: Array<out R>, transform: (a: UShort, b: R) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UIntArray.zip(other: Iterable<R>): List<Pair<UInt, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun <R> ULongArray.zip(other: Iterable<R>): List<Pair<ULong, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UByteArray.zip(other: Iterable<R>): List<Pair<UByte, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UShortArray.zip(other: Iterable<R>): List<Pair<UShort, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> UIntArray.zip(other: Iterable<R>, transform: (a: UInt, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> ULongArray.zip(other: Iterable<R>, transform: (a: ULong, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> UByteArray.zip(other: Iterable<R>, transform: (a: UByte, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> UShortArray.zip(other: Iterable<R>, transform: (a: UShort, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n *

Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun UIntArray.zip(other: UIntArray): List<Pair<UInt, UInt>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun ULongArray.zip(other: ULongArray): List<Pair<ULong, ULong>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun UByteArray.zip(other: UByteArray): List<Pair<UByte, UByte>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic infix fun UShortArray.zip(other: UShortArray): List<Pair<UShort, UShort>> {\n   return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> UIntArray.zip(other: UIntArray, transform: (a: UInt, b: UInt) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> ULongArray.zip(other: ULongArray, transform: (a: ULong, b: ULong) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> UByteArray.zip(other: UByteArray, transform: (a: UByte, b: UByte) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> UShortArray.zip(other: UShortArray, transform: (a: UShort, b: UShort) -> V): List<V> {\n   val size = minOf(size, other.size)\n   val list = ArrayList<V>(size)\n   for (i in 0 until size) {\n   list.add(transform(this[i], other[i]))\n   }\n   return list\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Array<out UInt>.sum(): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n

*/\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsigned Types::class)\npublic fun Array<out ULong>.sum(): ULong {\n   var sum: ULong = 0uL\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName(\"sumOfUByte\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsigned Types::class)\npublic fun Array<out UByte>.sum(): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName(\"sumOfUShort\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsigned Types::class)\npublic fun Array<out UShort>.sum(): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sum(): UInt {\n   return storage.sum().toUInt()\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sum(): ULong {\n   return storage.sum().toULong()\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sum(): UInt {\n   return sumOf { it.toUInt() }\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sum(): UInt {\n   return sumOf { it.toUInt() }\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"UCollectionsKt\")\n\npackage kotlin.collections\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\nimport kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns an array of UByte containing all of the elements of this collection.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Collection<UByte>.toUByteArray(): UByteArray {\n   val result = UByteArray(size)\n   var index = 0\n   for (element in this)\n   result[index++] = element\n   return result\n}\n\n/**\n * Returns an array of UInt containing all of the elements of this collection.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Collection<UInt>.toUIntArray(): UIntArray {\n   val result = UIntArray(size)\n   var index = 0\n   for (element in this)\n   result[index++] = element\n   return result\n}\n\n/**\n * Returns an array of ULong containing all of the elements of this collection.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Collection<ULong>.toULongArray(): ULongArray {\n   val result = ULongArray(size)\n   var index = 0\n   for (element in this)\n   result[index++] = element\n   return result\n}\n\n/**\n * Returns an array of UShort containing all of the elements of this collection.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Collection<UShort>.toUShortArray(): UShortArray {\n   val result = UShortArray(size)\n   var index = 0\n   for (element in this)\n   result[index++] = element\n   return result\n}\n\n/**\n * Returns the sum of all elements in the collection.\n */\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Iterable<UInt>.sum(): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n */\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsigned Types::class)\npublic fun Iterable<ULong>.sum(): ULong {\n   var sum: ULong = 0uL\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n */\n@kotlin.jvm.JvmName(\"sumOfUByte\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsigned Types::class)\npublic fun Iterable<UByte>.sum(): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n */\n@kotlin.jvm.JvmName(\"sumOfUShort\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsigned Types::class)\npublic fun Iterable<UShort>.sum(): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n sum += element\n   }\n   return sum\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming

Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
/\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"UComparisonsKt\")\n\npackage kotlin.comparisons\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.\n\n/**\n * Returns the greater of two values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun maxOf(a: UInt, b: UInt): UInt {\n   return if (a >= b) a else b\n}\n\n/**\n * Returns the greater of two values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun maxOf(a: ULong, b: ULong): ULong {\n   return if (a >= b) a else b\n}\n\n/**\n * Returns the greater of two values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun maxOf(a: UByte, b: UByte): UByte {\n   return if (a >= b) a else b\n}\n\n/**\n * Returns the greater of two values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun maxOf(a: UShort, b: UShort): UShort {\n   return if (a >= b) a else b\n}\n\n/**\n * Returns the greater of three values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun maxOf(a: UInt, b: UInt, c: UInt): UInt {\n   return maxOf(a, maxOf(b, c))\n}\n\n/**\n * Returns the greater of three values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun maxOf(a: ULong, b: ULong, c: ULong): ULong {\n   return maxOf(a, maxOf(b, c))\n}\n\n/**\n * Returns the greater of three values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun maxOf(a: UByte, b: UByte, c: UByte): UByte {\n   return maxOf(a, maxOf(b, c))\n}\n\n/**\n * Returns the greater of three values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun maxOf(a: UShort, b: UShort, c: UShort): UShort {\n   return maxOf(a, maxOf(b, c))\n}\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun maxOf(a: UInt, vararg other: UInt): UInt {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the greater of the given values.\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun maxOf(a: ULong, vararg other: ULong): ULong {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun maxOf(a: UByte, vararg other: UByte): UByte {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the greater of the given values.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun maxOf(a: UShort, vararg other: UShort): UShort {\n   var max = a\n   for (e in other) max = maxOf(max, e)\n   return max\n}\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun minOf(a: UInt, b: UInt): UInt {\n   return if (a <= b) a else b\n}\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun minOf(a: ULong, b: ULong): ULong {\n   return if (a <= b) a else b\n}\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun minOf(a: UByte, b: UByte): UByte {\n   return if (a <= b) a else b\n}\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun minOf(a: UShort, b: UShort): UShort {\n   return if (a <= b) a else b\n}\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun minOf(a: UInt, b: UInt, c: UInt): UInt {\n   return minOf(a, minOf(b, c))\n}\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun minOf(a: ULong, b: ULong, c: ULong): ULong {\n   return minOf(a, minOf(b, c))\n}\n\n/**\n *

Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\
npublic inline fun minOf(a: UByte, b: UByte, c: UByte): UByte {\n   return minOf(a, minOf(b, c))\n}\n\n/**\n *
Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\
npublic inline fun minOf(a: UShort, b: UShort, c: UShort): UShort {\n   return minOf(a, minOf(b, c))\n}\n\n/**\n *
Returns the smaller of the given values.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun
minOf(a: UInt, vararg other: UInt): UInt {\n   var min = a\n   for (e in other) min = minOf(min, e)\n   return
min\n}\n\n/**\n * Returns the smaller of the given values.\n
*/\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun minOf(a: ULong, vararg other: ULong):
ULong {\n   var min = a\n   for (e in other) min = minOf(min, e)\n   return min\n}\n\n/**\n * Returns the smaller
of the given values.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun minOf(a: UByte,
vararg other: UByte): UByte {\n   var min = a\n   for (e in other) min = minOf(min, e)\n   return min\n}\n\n/**\n *
Returns the smaller of the given values.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalUnsignedTypes\npublic fun
minOf(a: UShort, vararg other: UShort): UShort {\n   var min = a\n   for (e in other) min = minOf(min, e)\n
return min\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n *
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*/\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"URangesKt\")\n\npackage
kotlin.ranges\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\n\n/**\n * Returns a
random element from this range.\n * \n * @throws IllegalArgumentException if this range is empty.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\
npublic inline fun UIntRange.random(): UInt {\n   return random(Random)\n}\n\n/**\n * Returns a random element
from this range.\n * \n * @throws IllegalArgumentException if this range is empty.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\
npublic inline fun ULongRange.random(): ULong {\n   return random(Random)\n}\n\n/**\n * Returns a random
element from this range using the specified source of randomness.\n * \n * @throws IllegalArgumentException if
this range is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic
fun UIntRange.random(random: Random): UInt {\n   try {\n   return random.nextUInt(this)\n   } catch(e:
IllegalArgumentException) {\n   throw NoSuchElementException(e.message)\n   }\n}\n\n/**\n * Returns a
random element from this range using the specified source of randomness.\n * \n * @throws
IllegalArgumentException if this range is empty.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun
ULongRange.random(random: Random): ULong {\n   try {\n   return random.nextULong(this)\n   } catch(e:
IllegalArgumentException) {\n   throw NoSuchElementException(e.message)\n   }\n}\n\n/**\n * Returns a
random element from this range, or `null` if this range is empty.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class,
ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UIntRange.randomOrNull():
UInt? {\n   return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this range, or `null` if this
range is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class,
ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULongRange.randomOrNull():
ULong? {\n   return randomOrNull(Random)\n}\n\n/**\n * Returns a random element from this range using the
specified source of randomness, or `null` if this range is empty.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class,
ExperimentalUnsignedTypes::class)\npublic fun UIntRange.randomOrNull(random: Random): UInt? {\n   if
(isEmpty())\n   return null\n   return random.nextUInt(this)\n}\n\n/**\n * Returns a random element from this
range using the specified source of randomness, or `null` if this range is empty.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class,

ExperimentalUnsignedTypes::class)\npublic fun ULongRange.randomOrNull(random: Random): ULong? {\n if (isEmpty())\n return null\n return random.nextULong(this)\n}\n\n/**\n * Returns `true` if this range contains the specified [element].\n * \n * Always returns `false` if the [element] is `null`.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntRange.contains(element: UInt?): Boolean {\n return element != null && contains(element)\n}\n\n/**\n * Returns `true` if this range contains the specified [element].\n * \n * Always returns `false` if the [element] is `null`.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongRange.contains(element: ULong?): Boolean {\n return element != null && contains(element)\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic operator fun UIntRange.contains(value: UByte): Boolean {\n return contains(value.toUInt())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic operator fun ULongRange.contains(value: UByte): Boolean {\n return contains(value.toULong())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic operator fun ULongRange.contains(value: UInt): Boolean {\n return contains(value.toULong())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic operator fun UIntRange.contains(value: ULong): Boolean {\n return (value shr UInt.SIZE_BITS) == 0uL && contains(value.toUInt())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic operator fun UIntRange.contains(value: UShort): Boolean {\n return contains(value.toUInt())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic operator fun ULongRange.contains(value: UShort): Boolean {\n return contains(value.toULong())\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic infix fun UByte.downTo(to: UByte): UIntProgression {\n return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic infix fun UInt.downTo(to: UInt): UIntProgression {\n return UIntProgression.fromClosedRange(this, to, -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic infix fun ULong.downTo(to: ULong): ULongProgression {\n return ULongProgression.fromClosedRange(this, to, -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic infix fun UShort.downTo(to: UShort): UIntProgression {\n return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)\n}\n\n/**\n * Returns a progression that goes over the same range in the opposite direction with the same step.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun UIntProgression.reversed(): UIntProgression {\n return UIntProgression.fromClosedRange(last, first, -

step)\n}\n\n/**\n * Returns a progression that goes over the same range in the opposite direction with the same step.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun ULongProgression.reversed(): ULongProgression {\n   return ULongProgression.fromClosedRange(last, first, -step)\n}\n\n/**\n * Returns a progression that goes over the same range with the given step.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic infix fun UIntProgression.step(step: Int): UIntProgression {\n   checkStepIsPositive(step > 0, step)\n   return UIntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}\n\n/**\n * Returns a progression that goes over the same range with the given step.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic infix fun ULongProgression.step(step: Long): ULongProgression {\n   checkStepIsPositive(step > 0, step)\n   return ULongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic infix fun UByte.until(to: UByte): UIntRange {\n   if (to <= UByte.MIN_VALUE) return UIntRange.EMPTY\n   return this.toUInt() .. (to - 1u).toUInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic infix fun UInt.until(to: UInt): UIntRange {\n   if (to <= UInt.MIN_VALUE) return UIntRange.EMPTY\n   return this .. (to - 1u).toUInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic infix fun ULong.until(to: ULong): ULongRange {\n   if (to <= ULong.MIN_VALUE) return ULongRange.EMPTY\n   return this .. (to - 1u).toULong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic infix fun UShort.until(to: UShort): UIntRange {\n   if (to <= UShort.MIN_VALUE) return UIntRange.EMPTY\n   return this.toUInt() .. (to - 1u).toUInt()\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeastUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun UInt.coerceAtLeast(minimumValue: UInt): UInt {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeastUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun ULong.coerceAtLeast(minimumValue: ULong): ULong {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeastUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun UByte.coerceAtLeast(minimumValue: UByte): UByte {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeastUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun UShort.coerceAtLeast(minimumValue: UShort): UShort {\n   return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this

value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun UInt.coerceAtMost(maximumValue: UInt): UInt {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun ULong.coerceAtMost(maximumValue: ULong): ULong {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun UByte.coerceAtMost(maximumValue: UByte): UByte {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun UShort.coerceAtMost(maximumValue: UShort): UShort {\n   return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceInUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun UInt.coerceIn(minimumValue: UInt, maximumValue: UInt): UInt {\n   if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n   if (this < minimumValue) return minimumValue\n   if (this > maximumValue) return maximumValue\n   return this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceInUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun ULong.coerceIn(minimumValue: ULong, maximumValue: ULong): ULong {\n   if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n   if (this < minimumValue) return minimumValue\n   if (this > maximumValue) return maximumValue\n   return this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceInUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun UByte.coerceIn(minimumValue: UByte, maximumValue: UByte): UByte {\n   if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n   if (this < minimumValue) return minimumValue\n   if (this > maximumValue) return maximumValue\n   return this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceInUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun

UShort.coerceIn(minimumValue: UShort, maximumValue: UShort): UShort {\n   if (minimumValue > maximumValue) throw IllegalArgumentException(\"Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.\")\n   if (this < minimumValue) return minimumValue\n   if (this > maximumValue) return maximumValue\n   return this\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * \n * @return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.\n * \n * @sample samples.comparisons.ComparableOps.coerceInUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun UInt.coerceIn(range: ClosedRange<UInt>): UInt {\n   if (range is ClosedFloatingPointRange) {\n   return this.coerceIn<UInt>(range)\n   }\n   if (range.isEmpty()) throw IllegalArgumentException(\"Cannot coerce value to an empty range: $range.\")\n   return when {\n   this < range.start -> range.start\n   this > range.endInclusive -> range.endInclusive\n   else -> this\n   }\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * \n * @return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.\n * \n * @sample samples.comparisons.ComparableOps.coerceInUnsigned\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun ULong.coerceIn(range: ClosedRange<ULong>): ULong {\n   if (range is ClosedFloatingPointRange) {\n   return this.coerceIn<ULong>(range)\n   }\n   if (range.isEmpty()) throw IllegalArgumentException(\"Cannot coerce value to an empty range: $range.\")\n   return when {\n   this < range.start -> range.start\n   this > range.endInclusive -> range.endInclusive\n   else -> this\n   }\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"USequencesKt\")\n\npackage kotlin.sequences\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"sumOfUInt\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Sequence<UInt>.sum(): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"sumOfULong\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Sequence<ULong>.sum(): ULong {\n   var sum: ULong = 0uL\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"sumOfUByte\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Sequence<UByte>.sum(): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName(\"sumOfUShort\")\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Sequence<UShort>.sum(): UInt {\n   var sum: UInt = 0u\n   for (element in this) {\n   sum += element\n   }\n   return sum\n}\n\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\n\npublic expect open class Error : Throwable {\n   constructor()\n   constructor(message: String?)\n   constructor(message: String?, cause: Throwable?)\n   constructor(cause: Throwable?)\n}\n\npublic expect open class Exception : Throwable {\n   constructor()\n   constructor(message: String?)\n   constructor(message: String?, cause: Throwable?)\n   constructor(cause: Throwable?)\n}\n\npublic expect open class RuntimeException : Exception {\n   constructor()\n   constructor(message: String?)\n   constructor(message: String?, cause: Throwable?)\n   constructor(cause:

Throwable?)\n}\n\npublic expect open class IllegalArgumentException : RuntimeException {\n constructor()\n constructor(message: String?)\n constructor(message: String?, cause: Throwable?)\n constructor(cause: Throwable?)\n}\n\npublic expect open class IllegalStateException : RuntimeException {\n constructor()\n constructor(message: String?)\n constructor(message: String?, cause: Throwable?)\n constructor(cause: Throwable?)\n}\n\npublic expect open class IndexOutOfBoundsException : RuntimeException {\n constructor()\n constructor(message: String?)\n}\n\npublic expect open class ConcurrentModificationException : RuntimeException {\n constructor()\n constructor(message: String?)\n @Deprecated(\"The constructor is not supported on all platforms and will be removed from kotlin-stdlib-common soon.\", level = DeprecationLevel.ERROR)\n constructor(message: String?, cause: Throwable?)\n @Deprecated(\"The constructor is not supported on all platforms and will be removed from kotlin-stdlib-common soon.\", level = DeprecationLevel.ERROR)\n constructor(cause: Throwable?)\n}\n\npublic expect open class UnsupportedOperationException : RuntimeException {\n constructor()\n constructor(message: String?)\n constructor(message: String?, cause: Throwable?)\n constructor(cause: Throwable?)\n}\n\npublic expect open class NumberFormatException : IllegalArgumentException {\n constructor()\n constructor(message: String?)\n}\n\npublic expect open class NullPointerException : RuntimeException {\n constructor()\n constructor(message: String?)\n}\n\npublic expect open class ClassCastException : RuntimeException {\n constructor()\n constructor(message: String?)\n}\n\npublic expect open class AssertionError : Error {\n constructor()\n constructor(message: Any?)\n}\n\npublic expect open class NoSuchElementException : RuntimeException {\n constructor()\n constructor(message: String?)\n}\n\n@SinceKotlin(\"1.3\")\npublic expect open class ArithmeticException : RuntimeException {\n constructor()\n constructor(message: String?)\n}\n\n@Deprecated(\"This exception type is not supposed to be thrown or caught in common code and will be removed from kotlin-stdlib-common soon.\", level = DeprecationLevel.ERROR)\npublic expect open class NoWhenBranchMatchedException : RuntimeException {\n constructor()\n constructor(message: String?)\n constructor(message: String?, cause: Throwable?)\n constructor(cause: Throwable?)\n}\n\n@Deprecated(\"This exception type is not supposed to be thrown or caught in common code and will be removed from kotlin-stdlib-common soon.\", level = DeprecationLevel.ERROR)\npublic expect class UninitializedPropertyAccessException : RuntimeException {\n constructor()\n constructor(message: String?)\n constructor(message: String?, cause: Throwable?)\n constructor(cause: Throwable?)\n}\n\n/**\n * Thrown after invocation of a function or property that was expected to return `Nothing`, but returned something instead.\n */\n@SinceKotlin(\"1.4\")\n@PublishedApi\ninternal class KotlinNothingValueException : RuntimeException {\n constructor() : super()\n constructor(message: String?) : super(message)\n constructor(message: String?, cause: Throwable?) : super(message, cause)\n constructor(cause: Throwable?) : super(cause)\n}\n\n/**\n * Returns the detailed description of this throwable with its stack trace.\n *\n * The detailed description includes:\n * - the short description (see [Throwable.toString]) of this throwable;\n * - the complete stack trace;\n * - detailed descriptions of the exceptions that were [suppressed][suppressedExceptions] in order to deliver this exception;\n * - the detailed description of each throwable in the [Throwable.cause] chain.\n */\n@SinceKotlin(\"1.4\")\npublic expect fun Throwable.stackTraceToString(): String\n\n/**\n * Prints the [detailed description][Throwable.stackTraceToString] of this throwable to the standard output or standard error output.\n */\n@SinceKotlin(\"1.4\")\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\")\npublic expect fun Throwable.printStackTrace(): Unit\n\n/**\n * When supported by the platform, adds the specified exception to the list of exceptions that were\n * suppressed in order to deliver this exception.\n */\n@SinceKotlin(\"1.4\")\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\")\npublic expect fun Throwable.addSuppressed(exception: Throwable)\n\n/**\n * Returns a list of all exceptions that were suppressed in order to deliver this exception.\n *\n * The list can be empty:\n * - if no exceptions were suppressed;\n * - if the platform doesn't support suppressed exceptions;\n * - if this [Throwable] instance has disabled the suppression.\n */\n@SinceKotlin(\"1.4\")\npublic expect val Throwable.suppressedExceptions: List<Throwable>\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage

kotlin.js\n\nimport kotlin.annotation.AnnotationTarget.*\n\n/**\n * Gives a declaration (a function, a property or a class) specific name in JavaScript.\n */\n@Target(CLASS, FUNCTION, PROPERTY, CONSTRUCTOR, PROPERTY_GETTER, PROPERTY_SETTER)\n@OptionalExpectation\npublic expect annotation class JsName(val name: String)\n\n/**\n * Marks experimental JS export annotations.\n *\n * Note that behavior of these annotations will likely be changed in the future.\n *\n * Usages of such annotations will be reported as warnings unless an explicit opt-in with\n * the [OptIn] annotation, e.g. `@OptIn(ExperimentalJsExport::class)`,\n * or with the `-opt-in=kotlin.js.ExperimentalJsExport` compiler option is given.\n */\n@RequiresOptIn(level = RequiresOptIn.Level.WARNING)\n@MustBeDocumented\n@Retention(AnnotationRetention.BINARY)\n@SinceKotlin(\"1.4\")\npublic annotation class ExperimentalJsExport\n\n/**\n * Exports top-level declaration on JS platform.\n *\n * Compiled module exposes declarations that are marked with this annotation without name mangling.\n *\n * This annotation can be applied to either files or top-level declarations.\n *\n * It is currently prohibited to export the following kinds of declarations:\n *\n * * `expect` declarations\n * * inline functions with reified type parameters\n * * suspend functions\n * * secondary constructors without `@JsName`\n * * extension properties\n * * enum classes\n * * annotation classes\n *\n * Signatures of exported declarations must only contain \"exportable\" types:\n *\n * * `dynamic`, `Any`, `String`, `Boolean`, `Byte`, `Short`, `Int`, `Float`, `Double`\n * * `BooleanArray`, `ByteArray`, `ShortArray`, `IntArray`, `FloatArray`, `DoubleArray`\n * * `Array<exportable-type>`\n * * Function types with exportable parameters and return types\n * * `external` or `@JsExport` classes and interfaces\n * * Nullable counterparts of types above\n * * Unit return type. Must not be nullable\n *\n * This annotation is experimental, meaning that restrictions mentioned above are subject to change.\n */\n@ExperimentalJsExport\n@Retention(AnnotationRetention.BINARY)\n@Target(CLASS, PROPERTY, FUNCTION, FILE)\n@SinceKotlin(\"1.4\")\n@OptionalExpectation\npublic expect annotation class JsExport()","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.io\n\n\n/** Prints the line separator to the standard output stream. */\npublic expect fun println()\n\n/** Prints the given [message] and the line separator to the standard output stream. */\npublic expect fun println(message: Any?)\n\n/** Prints the given [message] to the standard output stream. */\npublic expect fun print(message: Any?)\n\n/**\n * Reads a line of input from the standard input stream and returns it,\n * or throws a [RuntimeException] if EOF has already been reached when [readln] is called.\n *\n * LF or CRLF is treated as the line terminator. Line terminator is not included in the returned string.\n *\n * Currently this function is not supported in Kotlin/JS and throws [UnsupportedOperationException].\n */\n@SinceKotlin(\"1.6\")\npublic expect fun readln(): String\n\n/**\n * Reads a line of input from the standard input stream and returns it,\n * or return `null` if EOF has already been reached when [readlnOrNull] is called.\n *\n * LF or CRLF is treated as the line terminator. Line terminator is not included in the returned string.\n *\n * Currently this function is not supported in Kotlin/JS and throws [UnsupportedOperationException].\n */\n@SinceKotlin(\"1.6\")\npublic expect fun readlnOrNull(): String?\n\ninternal class ReadAfterEOFException(message: String?) : RuntimeException(message)\n\n\ninternal expect interface Serializable\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\nimport kotlin.internal.PlatformDependent\n\n/**\n * Classes that inherit from this interface can be represented as a sequence of elements that can\n * be iterated over.\n * @param T the type of element being iterated over. The iterator is covariant in its element type.\n */\npublic interface Iterable<out T> {\n   /**\n   * Returns an iterator over the elements of this object.\n   */\n   public operator fun iterator(): Iterator<T>\n}\n\n/**\n * Classes that inherit from this interface can be represented as a sequence of elements that can\n * be iterated over and that supports removing elements during iteration.\n * @param T the type of element being iterated over. The mutable iterator is invariant in its element type.\n */\npublic interface MutableIterable<out T> : Iterable<T> {\n   /**\n   * Returns an iterator over the elements of this sequence that supports removing elements during iteration.\n   */\n   override fun iterator(): MutableIterator<T>\n}\n\n/**\n * A generic collection of elements. Methods in this interface support only read-only access to the collection;\n * read/write access is supported through the [MutableCollection] interface.\n * @param E the type of elements

```
contained in the collection. The collection is covariant in its element type.\n */\npublic interface Collection<out E> : Iterable<E> {\n    // Query Operations\n    /**\n     * Returns the size of the collection.\n     */\n    public val size: Int\n\n    /**\n     * Returns `true` if the collection is empty (contains no elements), `false` otherwise.\n     */\n    public fun isEmpty(): Boolean\n\n    /**\n     * Checks if the specified element is contained in this collection.\n     */\n    public operator fun contains(element: @UnsafeVariance E): Boolean\n\n    override fun iterator(): Iterator<E>\n\n    // Bulk Operations\n    /**\n     * Checks if all elements in the specified collection are contained in this collection.\n     */\n    public fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean\n}\n\n/**\n * A generic collection of elements that supports adding and removing elements.\n *\n * @param E the type of elements contained in the collection. The mutable collection is invariant in its element type.\n */\npublic interface MutableCollection<E> : Collection<E>, MutableIterable<E> {\n    // Query Operations\n    override fun iterator(): MutableIterator<E>\n\n    // Modification Operations\n    /**\n     * Adds the specified element to the collection.\n     *\n     * @return `true` if the element has been added, `false` if the collection does not support duplicates\n     * and the element is already contained in the collection.\n     */\n    public fun add(element: E): Boolean\n\n    /**\n     * Removes a single instance of the specified element from this\n     * collection, if it is present.\n     *\n     * @return `true` if the element has been successfully removed; `false` if it was not present in the collection.\n     */\n    public fun remove(element: E): Boolean\n\n    // Bulk Modification Operations\n    /**\n     * Adds all of the elements of the specified collection to this collection.\n     *\n     * @return `true` if any of the specified elements was added to the collection, `false` if the collection was not modified.\n     */\n    public fun addAll(elements: Collection<E>): Boolean\n\n    /**\n     * Removes all of this collection's elements that are also contained in the specified collection.\n     *\n     * @return `true` if any of the specified elements was removed from the collection, `false` if the collection was not modified.\n     */\n    public fun removeAll(elements: Collection<E>): Boolean\n\n    /**\n     * Retains only the elements in this collection that are contained in the specified collection.\n     *\n     * @return `true` if any element was removed from the collection, `false` if the collection was not modified.\n     */\n    public fun retainAll(elements: Collection<E>): Boolean\n\n    /**\n     * Removes all elements from this collection.\n     */\n    public fun clear(): Unit\n}\n\n/**\n * A generic ordered collection of elements. Methods in this interface support only read-only access to the list;\n * read/write access is supported through the [MutableList] interface.\n * @param E the type of elements contained in the list. The list is covariant in its element type.\n */\npublic interface List<out E> : Collection<E> {\n    // Query Operations\n\n    override val size: Int\n    override fun isEmpty(): Boolean\n    override fun contains(element: @UnsafeVariance E): Boolean\n    override fun iterator(): Iterator<E>\n\n    // Bulk Operations\n    override fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean\n\n    // Positional Access Operations\n    /**\n     * Returns the element at the specified index in the list.\n     */\n    public operator fun get(index: Int): E\n\n    // Search Operations\n    /**\n     * Returns the index of the first occurrence of the specified element in the list, or -1 if the specified\n     * element is not contained in the list.\n     */\n    public fun indexOf(element: @UnsafeVariance E): Int\n\n    /**\n     * Returns the index of the last occurrence of the specified element in the list, or -1 if the specified\n     * element is not contained in the list.\n     */\n    public fun lastIndexOf(element: @UnsafeVariance E): Int\n\n    // List Iterators\n    /**\n     * Returns a list iterator over the elements in this list (in proper sequence).\n     */\n    public fun listIterator(): ListIterator<E>\n\n    /**\n     * Returns a list iterator over the elements in this list (in proper sequence), starting at the specified [index].\n     */\n    public fun listIterator(index: Int): ListIterator<E>\n\n    // View\n    /**\n     * Returns a view of the portion of this list between the specified [fromIndex] (inclusive) and [toIndex] (exclusive).\n     * The returned list is backed by this list, so non-structural changes in the returned list are reflected in this list, and vice-versa.\n     *\n     * Structural changes in the base list make the behavior of the view undefined.\n     */\n    public fun subList(fromIndex: Int, toIndex: Int): List<E>\n}\n\n/**\n * A generic ordered collection of elements that supports adding and removing elements.\n * @param E the type of elements contained in the list. The mutable list is invariant in its element type.\n */\npublic interface MutableList<E> : List<E>, MutableCollection<E> {\n    // Modification Operations\n    /**\n     * Adds the specified element to the end of this list.\n     *\n     * @return `true` because the list is always modified as the result of this operation.\n     */\n    override fun add(element: E): Boolean\n\n    override fun remove(element: E): Boolean\n\n    // Bulk Modification Operations\n    /**\n     * Adds all of the
```

elements of the specified collection to the end of this list.\n *\n * The elements are appended in the order they appear in the [elements] collection.\n *\n * @return `true` if the list was changed as the result of the operation.\n */\n override fun addAll(elements: Collection<E>): Boolean\n\n /**\n * Inserts all of the elements of the specified collection [elements] into this list at the specified [index].\n *\n * @return `true` if the list was changed as the result of the operation.\n */\n public fun addAll(index: Int, elements: Collection<E>): Boolean\n\n override fun removeAll(elements: Collection<E>): Boolean\n override fun retainAll(elements: Collection<E>): Boolean\n override fun clear(): Unit\n\n // Positional Access Operations\n /**\n * Replaces the element at the specified position in this list with the specified element.\n *\n * @return the element previously at the specified position.\n */\n public operator fun set(index: Int, element: E): E\n\n /**\n * Inserts an element into the list at the specified [index].\n */\n public fun add(index: Int, element: E): Unit\n\n /**\n * Removes an element at the specified [index] from the list.\n *\n * @return the element that has been removed.\n */\n public fun removeAt(index: Int): E\n\n // List Iterators\n override fun listIterator(): MutableListIterator<E>\n\n override fun listIterator(index: Int): MutableListIterator<E>\n\n // View\n override fun subList(fromIndex: Int, toIndex: Int): MutableList<E>\n}\n\n/**\n * A generic unordered collection of elements that does not support duplicate elements.\n * Methods in this interface support only read-only access to the set;\n * read/write access is supported through the [MutableSet] interface.\n * @param E the type of elements contained in the set. The set is covariant in its element type.\n */\npublic interface Set<out E> : Collection<E> {\n // Query Operations\n\n override val size: Int\n override fun isEmpty(): Boolean\n override fun contains(element: @UnsafeVariance E): Boolean\n override fun iterator(): Iterator<E>\n\n // Bulk Operations\n override fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean\n}\n\n/**\n * A generic unordered collection of elements that does not support duplicate elements, and supports\n * adding and removing elements.\n * @param E the type of elements contained in the set. The mutable set is invariant in its element type.\n */\npublic interface MutableSet<E> : Set<E>, MutableCollection<E> {\n // Query Operations\n override fun iterator(): MutableIterator<E>\n\n // Modification Operations\n\n /**\n * Adds the specified element to the set.\n *\n * @return `true` if the element has been added, `false` if the element is already contained in the set.\n */\n override fun add(element: E): Boolean\n\n override fun remove(element: E): Boolean\n\n // Bulk Modification Operations\n\n override fun addAll(elements: Collection<E>): Boolean\n override fun removeAll(elements: Collection<E>): Boolean\n override fun retainAll(elements: Collection<E>): Boolean\n override fun clear(): Unit\n}\n\n/**\n * A collection that holds pairs of objects (keys and values) and supports efficiently retrieving\n * the value corresponding to each key. Map keys are unique; the map holds only one value for each key.\n * Methods in this interface support only read-only access to the map; read-write access is supported through\n * the [MutableMap] interface.\n * @param K the type of map keys. The map is invariant in its key type, as it\n * can accept key as a parameter (of [containsKey] for example) and return it in [keys] set.\n * @param V the type of map values. The map is covariant in its value type.\n */\npublic interface Map<K, out V> {\n // Query Operations\n /**\n * Returns the number of key/value pairs in the map.\n */\n public val size: Int\n\n /**\n * Returns `true` if the map is empty (contains no elements), `false` otherwise.\n */\n public fun isEmpty(): Boolean\n\n /**\n * Returns `true` if the map contains the specified [key].\n */\n public fun containsKey(key: K): Boolean\n\n /**\n * Returns `true` if the map maps one or more keys to the specified [value].\n */\n public fun containsValue(value: @UnsafeVariance V): Boolean\n\n /**\n * Returns the value corresponding to the given [key], or `null` if such a key is not present in the map.\n */\n public operator fun get(key: K): V?\n\n /**\n * Returns the value corresponding to the given [key], or [defaultValue] if such a key is not present in the map.\n *\n * @since JDK 1.8\n */\n @SinceKotlin(\"1.1\")\n @PlatformDependent\n public fun getOrDefault(key: K, defaultValue: @UnsafeVariance V): V {\n // See default implementation in JDK sources\n throw NotImplementedError()\n }\n\n // Views\n /**\n * Returns a read-only [Set] of all keys in this map.\n */\n public val keys: Set<K>\n\n /**\n * Returns a read-only [Collection] of all values in this map. Note that this collection may contain duplicate values.\n */\n public val values: Collection<V>\n\n /**\n * Returns a read-only [Set] of all key/value pairs in this map.\n */\n public val entries: Set<Map.Entry<K, V>>\n\n /**\n * Represents a key/value pair held by a [Map].\n */\n public interface Entry<out K, out V>

{\n /**\n * Returns the key of this key/value pair.\n */\n public val key: K\n\n /**\n * Returns the value of this key/value pair.\n */\n public val value: V\n }\n}\n\n/**\n * A modifiable collection that holds pairs of objects (keys and values) and supports efficiently retrieving\n * the value corresponding to each key. Map keys are unique; the map holds only one value for each key.\n * @param K the type of map keys. The map is invariant in its key type.\n * @param V the type of map values. The mutable map is invariant in its value type.\n */\npublic interface MutableMap<K, V> : Map<K, V> {\n // Modification Operations\n /**\n * Associates the specified [value] with the specified [key] in the map.\n *\n * @return the previous value associated with the key, or `null` if the key was not present in the map.\n */\n public fun put(key: K, value: V): V?\n\n /**\n * Removes the specified key and its corresponding value from this map.\n *\n * @return the previous value associated with the key, or `null` if the key was not present in the map.\n */\n public fun remove(key: K): V?\n\n /**\n * Removes the entry for the specified key only if it is mapped to the specified value.\n *\n * @return true if entry was removed\n */\n @SinceKotlin(\"1.1\")\n @PlatformDependent\n public fun remove(key: K, value: V): Boolean {\n // See default implementation in JDK sources\n return true\n }\n\n // Bulk Modification Operations\n /**\n * Updates this map with key/value pairs from the specified map [from].\n */\n public fun putAll(from: Map<out K, V>): Unit\n\n /**\n * Removes all elements from this map.\n */\n public fun clear(): Unit\n\n // Views\n /**\n * Returns a [MutableSet] of all keys in this map.\n */\n override val keys: MutableSet<K>\n\n /**\n * Returns a [MutableCollection] of all values in this map. Note that this collection may contain duplicate values.\n */\n override val values: MutableCollection<V>\n\n /**\n * Returns a [MutableSet] of all key/value pairs in this map.\n */\n override val entries: MutableSet<MutableMap.MutableEntry<K, V>>\n\n /**\n * Represents a key/value pair held by a [MutableMap].\n */\n public interface MutableEntry<K, V> : Map.Entry<K, V> {\n /**\n * Changes the value associated with the key of this entry.\n *\n * @return the previous value corresponding to the key.\n */\n public fun setValue(newValue: V): V\n }\n}\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin.collections\n\n/** An iterator over a sequence of values of type `Byte`. */\npublic abstract class ByteIterator : Iterator<Byte> {\n override final fun next() = nextByte()\n\n /** Returns the next value in the sequence without boxing. */\n public abstract fun nextByte(): Byte\n}\n\n/** An iterator over a sequence of values of type `Char`. */\npublic abstract class CharIterator : Iterator<Char> {\n override final fun next() = nextChar()\n\n /** Returns the next value in the sequence without boxing. */\n public abstract fun nextChar(): Char\n}\n\n/** An iterator over a sequence of values of type `Short`. */\npublic abstract class ShortIterator : Iterator<Short> {\n override final fun next() = nextShort()\n\n /** Returns the next value in the sequence without boxing. */\n public abstract fun nextShort(): Short\n}\n\n/** An iterator over a sequence of values of type `Int`. */\npublic abstract class IntIterator : Iterator<Int> {\n override final fun next() = nextInt()\n\n /** Returns the next value in the sequence without boxing. */\n public abstract fun nextInt(): Int\n}\n\n/** An iterator over a sequence of values of type `Long`. */\npublic abstract class LongIterator : Iterator<Long> {\n override final fun next() = nextLong()\n\n /** Returns the next value in the sequence without boxing. */\n public abstract fun nextLong(): Long\n}\n\n/** An iterator over a sequence of values of type `Float`. */\npublic abstract class FloatIterator : Iterator<Float> {\n override final fun next() = nextFloat()\n\n /** Returns the next value in the sequence without boxing. */\n public abstract fun nextFloat(): Float\n}\n\n/** An iterator over a sequence of values of type `Double`. */\npublic abstract class DoubleIterator : Iterator<Double> {\n override final fun next() = nextDouble()\n\n /** Returns the next value in the sequence without boxing. */\n public abstract fun nextDouble(): Double\n}\n\n/** An iterator over a sequence of values of type `Boolean`. */\npublic abstract class BooleanIterator : Iterator<Boolean> {\n override final fun next() = nextBoolean()\n\n /** Returns the next value in the sequence without boxing. */\n public abstract fun nextBoolean(): Boolean\n}\n\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin.ranges\n\n/**\n * An iterator over a progression of values of type `Char`.\n * @property

step the number by which the value is incremented on each step.\n */\ninternal class CharProgressionIterator(first: Char, last: Char, val step: Int) : CharIterator() {\n private val finalElement: Int = last.code\n private var hasNext: Boolean = if (step > 0) first <= last else first >= last\n private var next: Int = if (hasNext) first.code else finalElement\n\n override fun hasNext(): Boolean = hasNext\n\n override fun nextChar(): Char {\n val value = next\n if (value == finalElement) {\n if (!hasNext) throw kotlin.NoSuchElementException()\n hasNext = false\n }\n else {\n next += step\n }\n return value.toChar()\n }\n}\n\n/**\n * An iterator over a progression of values of type `Int`.\n * @property step the number by which the value is incremented on each step.\n */\ninternal class IntProgressionIterator(first: Int, last: Int, val step: Int) : IntIterator() {\n private val finalElement: Int = last\n private var hasNext: Boolean = if (step > 0) first <= last else first >= last\n private var next: Int = if (hasNext) first else finalElement\n\n override fun hasNext(): Boolean = hasNext\n\n override fun nextInt(): Int {\n val value = next\n if (value == finalElement) {\n if (!hasNext) throw kotlin.NoSuchElementException()\n hasNext = false\n }\n else {\n next += step\n }\n return value\n }\n}\n\n/**\n * An iterator over a progression of values of type `Long`.\n * @property step the number by which the value is incremented on each step.\n */\ninternal class LongProgressionIterator(first: Long, last: Long, val step: Long) : LongIterator() {\n private val finalElement: Long = last\n private var hasNext: Boolean = if (step > 0) first <= last else first >= last\n private var next: Long = if (hasNext) first else finalElement\n\n override fun hasNext(): Boolean = hasNext\n\n override fun nextLong(): Long {\n val value = next\n if (value == finalElement) {\n if (!hasNext) throw kotlin.NoSuchElementException()\n hasNext = false\n }\n else {\n next += step\n }\n return value\n }\n}\n\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin.ranges\n\nimport kotlin.internal.getProgressionLastElement\n\n/**\n * A progression of values of type `Char`.\n */\npublic open class CharProgression\n internal constructor\n (\n start: Char,\n endInclusive: Char,\n step: Int\n) : Iterable<Char> {\n init {\n if (step == 0) throw kotlin.IllegalArgumentException(\"Step must be non-zero.\")\n if (step == Int.MIN_VALUE) throw kotlin.IllegalArgumentException(\"Step must be greater than Int.MIN_VALUE to avoid overflow on negation.\")\n }\n\n /**\n * The first element in the progression.\n */\n public val first: Char = start\n\n /**\n * The last element in the progression.\n */\n public val last: Char = getProgressionLastElement(start.code, endInclusive.code, step).toChar()\n\n /**\n * The step of the progression.\n */\n public val step: Int = step\n\n override fun iterator(): CharIterator = CharProgressionIterator(first, last, step)\n\n /**\n * Checks if the progression is empty.\n *\n * Progression with a positive step is empty if its first element is greater than the last element.\n * Progression with a negative step is empty if its first element is less than the last element.\n */\n public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last\n\n override fun equals(other: Any?): Boolean =\n other is CharProgression && (isEmpty() && other.isEmpty() ||\n first == other.first && last == other.last && step == other.step)\n\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * (31 * first.code + last.code) + step)\n\n override fun toString(): String = if (step > 0) \"$first..$last step $step\" else \"$first downTo $last step ${-step}\"\n\n companion object {\n /**\n * Creates CharProgression within the specified bounds of a closed range.\n *\n * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].\n * In order to go backwards the [step] must be negative.\n *\n * [step] must be greater than `Int.MIN_VALUE` and not equal to zero.\n */\n public fun fromClosedRange(rangeStart: Char, rangeEnd: Char, step: Int): CharProgression = CharProgression(rangeStart, rangeEnd, step)\n }\n}\n\n/**\n * A progression of values of type `Int`.\n */\npublic open class IntProgression\n internal constructor\n (\n start: Int,\n endInclusive: Int,\n step: Int\n) : Iterable<Int> {\n init {\n if (step == 0) throw kotlin.IllegalArgumentException(\"Step must be non-zero.\")\n if (step == Int.MIN_VALUE) throw kotlin.IllegalArgumentException(\"Step must be greater than Int.MIN_VALUE to avoid overflow on negation.\")\n }\n\n /**\n * The first element in the progression.\n */\n public val first: Int = start\n\n /**\n * The last element in the progression.\n */\n public val last: Int =

getProgressionLastElement(start, endInclusive, step)\n\n   /**\n   * The step of the progression.\n   */\n   public val step: Int = step\n\n   override fun iterator(): IntIterator = IntProgressionIterator(first, last, step)\n\n   /**\n   * Checks if the progression is empty.\n   *\n   * Progression with a positive step is empty if its first element is greater than the last element.\n   * Progression with a negative step is empty if its first element is less than the last element.\n   */\n   public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last\n\n   override fun equals(other: Any?): Boolean =\n   other is IntProgression && (isEmpty() && other.isEmpty() ||\n   first == other.first && last == other.last && step == other.step)\n\n   override fun hashCode(): Int =\n   if (isEmpty()) -1 else (31 * (31 * first + last) + step)\n\n   override fun toString(): String = if (step > 0) \"$first..$last step $step\" else \"$first downTo $last step ${-step}\"\n\n   companion object {\n   /**\n   * Creates IntProgression within the specified bounds of a closed range.\n   *\n   * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].\n   * In order to go backwards the [step] must be negative.\n   *\n   * [step] must be greater than `Int.MIN_VALUE` and not equal to zero.\n   */\n   public fun fromClosedRange(rangeStart: Int, rangeEnd: Int, step: Int): IntProgression = IntProgression(rangeStart, rangeEnd, step)\n   }\n}\n\n/**\n * A progression of values of type `Long`.\n */\npublic open class LongProgression\n   internal constructor\n   (\n   start: Long,\n   endInclusive: Long,\n   step: Long\n   ) : Iterable<Long> {\n   init {\n   if (step == 0L) throw kotlin.IllegalArgumentException(\"Step must be non-zero.\")\n   if (step == Long.MIN_VALUE) throw kotlin.IllegalArgumentException(\"Step must be greater than Long.MIN_VALUE to avoid overflow on negation.\")\n   }\n\n   /**\n   * The first element in the progression.\n   */\n   public val first: Long = start\n\n   /**\n   * The last element in the progression.\n   */\n   public val last: Long = getProgressionLastElement(start, endInclusive, step)\n\n   /**\n   * The step of the progression.\n   */\n   public val step: Long = step\n\n   override fun iterator(): LongIterator = LongProgressionIterator(first, last, step)\n\n   /**\n   * Checks if the progression is empty.\n   *\n   * Progression with a positive step is empty if its first element is greater than the last element.\n   * Progression with a negative step is empty if its first element is less than the last element.\n   */\n   public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last\n\n   override fun equals(other: Any?): Boolean =\n   other is LongProgression && (isEmpty() && other.isEmpty() ||\n   first == other.first && last == other.last && step == other.step)\n\n   override fun hashCode(): Int =\n   if (isEmpty()) -1 else (31 * (31 * (first xor (first ushr 32)) + (last xor (last ushr 32))) + (step xor (step ushr 32))).toInt()\n\n   override fun toString(): String = if (step > 0) \"$first..$last step $step\" else \"$first downTo $last step ${-step}\"\n\n   companion object {\n   /**\n   * Creates LongProgression within the specified bounds of a closed range.\n   *\n   * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].\n   * In order to go backwards the [step] must be negative.\n   *\n   * [step] must be greater than `Long.MIN_VALUE` and not equal to zero.\n   */\n   public fun fromClosedRange(rangeStart: Long, rangeEnd: Long, step: Long): LongProgression = LongProgression(rangeStart, rangeEnd, step)\n   }\n}\n\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.ranges\n\n/**\n * Represents a range of values (for example, numbers or characters).\n * See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/ranges.html) for more information.\n */\npublic interface ClosedRange<T: Comparable<T>> {\n   /**\n   * The minimum value in the range.\n   */\n   public val start: T\n\n   /**\n   * The maximum value in the range (inclusive).\n   */\n   public val endInclusive: T\n\n   /**\n   * Checks whether the specified [value] belongs to the range.\n   */\n   public operator fun contains(value: T): Boolean = value >= start && value <= endInclusive\n\n   /**\n   * Checks whether the range is empty.\n   *\n   * The range is empty if its start value is greater than the end value.\n   */\n   public fun isEmpty(): Boolean = start > endInclusive\n}\n","/*\n * Copyright 2010-2015 JetBrains s.r.o.\n *\n * Licensed under the Apache License, Version 2.0 (the \"License\");\n * you may not use this file except in compliance with the License.\n * You may obtain a copy of the License at\n *\n * http://www.apache.org/licenses/LICENSE-2.0\n *\n * Unless required by applicable law or agreed to in writing, software\n * distributed under the License is distributed on an \"AS IS\" BASIS,\n * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.\n * See the

License for the specific language governing permissions and\n * limitations under the License.\n */\n\npackage kotlin\n\n/**\n * The type with only one value: the `Unit` object. This type corresponds to the `void` type in Java.\n */\npublic object Unit {\n override fun toString() = \"kotlin.Unit\"\n}\n","/*\n * Copyright 2010-2015 JetBrains s.r.o.\n *\n * Licensed under the Apache License, Version 2.0 (the \"License\");\n * you may not use this file except in compliance with the License.\n * You may obtain a copy of the License at\n *\n * http://www.apache.org/licenses/LICENSE-2.0\n *\n * Unless required by applicable law or agreed to in writing, software\n * distributed under the License is distributed on an \"AS IS\" BASIS,\n * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.\n * See the License for the specific language governing permissions and\n * limitations under the License.\n */\n\npackage kotlin.annotation\n\nimport kotlin.annotation.AnnotationTarget.*\n\n/**\n * Contains the list of code elements which are the possible annotation targets\n */\npublic enum class AnnotationTarget {\n /** Class, interface or object, annotation class is also included */\n CLASS,\n /** Annotation class only */\n ANNOTATION_CLASS,\n /** Generic type parameter */\n TYPE_PARAMETER,\n /** Property */\n PROPERTY,\n /** Field, including property's backing field */\n FIELD,\n /** Local variable */\n LOCAL_VARIABLE,\n /** Value parameter of a function or a constructor */\n VALUE_PARAMETER,\n /** Constructor only (primary or secondary) */\n CONSTRUCTOR,\n /** Function (constructors are not included) */\n FUNCTION,\n /** Property getter only */\n PROPERTY_GETTER,\n /** Property setter only */\n PROPERTY_SETTER,\n /** Type usage */\n TYPE,\n /** Any expression */\n EXPRESSION,\n /** File */\n FILE,\n /** Type alias */\n @SinceKotlin(\"1.1\")\n TYPEALIAS\n}\n\n/**\n * Contains the list of possible annotation's retentions.\n *\n * Determines how an annotation is stored in binary output.\n */\npublic enum class AnnotationRetention {\n /** Annotation isn't stored in binary output */\n SOURCE,\n /** Annotation is stored in binary output, but invisible for reflection */\n BINARY,\n /** Annotation is stored in binary output and visible for reflection (default retention) */\n RUNTIME\n}\n\n/**\n * This meta-annotation indicates the kinds of code elements which are possible targets of an annotation.\n *\n * If the target meta-annotation is not present on an annotation declaration, the annotation is applicable to the following elements:\n * [CLASS], [PROPERTY], [FIELD], [LOCAL_VARIABLE], [VALUE_PARAMETER], [CONSTRUCTOR], [FUNCTION], [PROPERTY_GETTER], [PROPERTY_SETTER].\n *\n * @property allowedTargets list of allowed annotation targets\n */\n@Target(AnnotationTarget.ANNOTATION_CLASS)\n@MustBeDocumented\npublic annotation class Target(vararg val allowedTargets: AnnotationTarget)\n\n/**\n * This meta-annotation determines whether an annotation is stored in binary output and visible for reflection. By default, both are true.\n *\n * @property value necessary annotation retention (RUNTIME, BINARY or SOURCE)\n */\n@Target(AnnotationTarget.ANNOTATION_CLASS)\npublic annotation class Retention(val value: AnnotationRetention = AnnotationRetention.RUNTIME)\n\n/**\n * This meta-annotation determines that an annotation is applicable twice or more on a single code element\n */\n@Target(AnnotationTarget.ANNOTATION_CLASS)\npublic annotation class Repeatable\n\n/**\n * This meta-annotation determines that an annotation is a part of public API and therefore should be included in the generated\n * documentation for the element to which the annotation is applied.\n */\n@Target(AnnotationTarget.ANNOTATION_CLASS)\npublic annotation class MustBeDocumented\n","/*\n * Copyright 2010-2016 JetBrains s.r.o.\n *\n * Licensed under the Apache License, Version 2.0 (the \"License\");\n * you may not use this file except in compliance with the License.\n * You may obtain a copy of the License at\n *\n * http://www.apache.org/licenses/LICENSE-2.0\n *\n * Unless required by applicable law or agreed to in writing, software\n * distributed under the License is distributed on an \"AS IS\" BASIS,\n * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.\n * See the License for the specific language governing permissions and\n * limitations under the License.\n */\n\npackage kotlin.internal\n\n/**\n * Specifies that the corresponding type parameter is not used for unsafe operations such as casts or 'is' checks\n * That means it's completely safe to use generic types as argument for such parameter.\n */\n@Target(AnnotationTarget.TYPE_PARAMETER)\n@Retention(AnnotationRetention.BINARY)\ninternal annotation class PureReifiable\n\n/**\n * Specifies that the corresponding built-in method exists depending on

platform.\n * Current implementation for JVM looks whether method with same JVM descriptor exists in the module JDK.\n * For example MutableMap.remove(K, V) available only if corresponding\n * method 'java/util/Map.remove(Ljava/lang/Object;Ljava/lang/Object;)Z' is defined in JDK (i.e. for major versions >= 8)\n */\n@Target(AnnotationTarget.FUNCTION)\n@Retention(AnnotationRetention.BINARY)\ninternal annotation class PlatformDependent\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.internal\n\n// a mod b (in arithmetical sense)\nprivate fun mod(a: Int, b: Int): Int {\n   val mod = a % b\n   return if (mod >= 0) mod else mod + b\n}\n\nprivate fun mod(a: Long, b: Long): Long {\n   val mod = a % b\n   return if (mod >= 0) mod else mod + b\n}\n\n// (a - b) mod c\nprivate fun differenceModulo(a: Int, b: Int, c: Int): Int {\n   return mod(mod(a, c) - mod(b, c), c)\n}\n\nprivate fun differenceModulo(a: Long, b: Long, c: Long): Long {\n   return mod(mod(a, c) - mod(b, c), c)\n}\n\n/**\n * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range\n * from [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative\n * [step].\n *\n * No validation on passed parameters is performed. The given parameters should satisfy the condition:\n *\n * - either `step > 0` and `start <= end`,\n * - or `step < 0` and `start >= end`.\n *\n * @param start first element of the progression\n * @param end ending bound for the progression\n * @param step increment, or difference of successive elements in the progression\n * @return the final element of the progression\n * @suppress\n */\n@PublishedApi\ninternal fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {\n   step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)\n   step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)\n   else -> throw kotlin.IllegalArgumentException(\"Step is zero.\")\n}\n\n/**\n * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range\n * from [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative\n * [step].\n *\n * No validation on passed parameters is performed. The given parameters should satisfy the condition:\n *\n * - either `step > 0` and `start <= end`,\n * - or `step < 0` and `start >= end`.\n *\n * @param start first element of the progression\n * @param end ending bound for the progression\n * @param step increment, or difference of successive elements in the progression\n * @return the final element of the progression\n * @suppress\n */\n@PublishedApi\ninternal fun getProgressionLastElement(start: Long, end: Long, step: Long): Long = when {\n   step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)\n   step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)\n   else -> throw kotlin.IllegalArgumentException(\"Step is zero.\")\n}\n\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@JsName(\"arrayIterator\")\ninternal fun arrayIterator(array: dynamic, type: String?) = when (type) {\n   null -> {\n   val arr: Array<dynamic> = array\n   object : Iterator<dynamic> {\n   var index = 0\n   override fun hasNext() = index < arr.size\n   override fun next() = if (index < arr.size) arr[index++] else throw NoSuchElementException(\"$index\")\n   }\n   }\n   \"BooleanArray\" -> booleanArrayIterator(array)\n   \"ByteArray\" -> byteArrayIterator(array)\n   \"ShortArray\" -> shortArrayIterator(array)\n   \"CharArray\" -> charArrayIterator(array)\n   \"IntArray\" -> intArrayIterator(array)\n   \"LongArray\" -> longArrayIterator(array)\n   \"FloatArray\" -> floatArrayIterator(array)\n   \"DoubleArray\" -> doubleArrayIterator(array)\n   else -> throw IllegalStateException(\"Unsupported type argument for arrayIterator: $type\")\n}\n\n@JsName(\"booleanArrayIterator\")\ninternal fun booleanArrayIterator(array: BooleanArray) = object : BooleanIterator() {\n   var index = 0\n   override fun hasNext() = index < array.size\n   override fun nextBoolean() = if (index < array.size) array[index++] else throw NoSuchElementException(\"$index\")\n}\n\n@JsName(\"byteArrayIterator\")\ninternal fun byteArrayIterator(array: ByteArray) = object : ByteIterator() {\n   var index = 0\n   override fun hasNext() = index < array.size\n   override fun nextByte() = if (index < array.size) array[index++] else throw NoSuchElementException(\"$index\")\n}\n\n@JsName(\"shortArrayIterator\")\ninternal fun shortArrayIterator(array: ShortArray) = object : ShortIterator() {\n   var index = 0\n   override fun hasNext() =

```
index < array.size\n    override fun nextShort() = if (index < array.size) array[index++] else throw NoSuchElementException(\"$index\")\n}\n\n@JsName(\"charArrayIterator\")\ninternal fun charArrayIterator(array: CharArray) = object : CharIterator() {\n    var index = 0\n    override fun hasNext() = index < array.size\n    override fun nextChar() = if (index < array.size) array[index++] else throw NoSuchElementException(\"$index\")\n}\n\n@JsName(\"intArrayIterator\")\ninternal fun intArrayIterator(array: IntArray) = object : IntIterator() {\n    var index = 0\n    override fun hasNext() = index < array.size\n    override fun nextInt() = if (index < array.size) array[index++] else throw NoSuchElementException(\"$index\")\n}\n\n@JsName(\"floatArrayIterator\")\ninternal fun floatArrayIterator(array: FloatArray) = object : FloatIterator() {\n    var index = 0\n    override fun hasNext() = index < array.size\n    override fun nextFloat() = if (index < array.size) array[index++] else throw NoSuchElementException(\"$index\")\n}\n\n@JsName(\"doubleArrayIterator\")\ninternal fun doubleArrayIterator(array: DoubleArray) = object : DoubleIterator() {\n    var index = 0\n    override fun hasNext() = index < array.size\n    override fun nextDouble() = if (index < array.size) array[index++] else throw NoSuchElementException(\"$index\")\n}\n\n@JsName(\"longArrayIterator\")\ninternal fun longArrayIterator(array: LongArray) = object : LongIterator() {\n    var index = 0\n    override fun hasNext() = index < array.size\n    override fun nextLong() = if (index < array.size) array[index++] else throw NoSuchElementException(\"$index\")\n}\n\n@JsName(\"PropertyMetadata\")\ninternal class PropertyMetadata(@JsName(\"callableName\") val name: String)\n\n@JsName(\"noWhenBranchMatched\")\ninternal fun noWhenBranchMatched(): Nothing = throw NoWhenBranchMatchedException()\n\n@JsName(\"subSequence\")\ninternal fun subSequence(c: CharSequence, startIndex: Int, endIndex: Int): CharSequence {\n    if (c is String) {\n        return c.substring(startIndex, endIndex)\n    } else {\n        return c.asDynamic().`subSequence_vux9f0$`(startIndex, endIndex)\n    }\n}\n\n@JsName(\"captureStack\")\ninternal fun captureStack(@Suppress(\"UNUSED_PARAMETER\") baseClass: JsClass<in Throwable>, instance: Throwable) {\n    if (js(\"Error\").captureStackTrace) {\n        // Using uncropped stack traces due to KT-37563.\n        // Precise stack traces are implemented in JS IR compiler and stdlib\n        js(\"Error\").captureStackTrace(instance);\n    } else {\n        instance.asDynamic().stack = js(\"new Error()\").stack;\n    }\n}\n\n@JsName(\"newThrowable\")\ninternal fun newThrowable(message: String?, cause: Throwable?): Throwable {\n    val throwable = js(\"new Error()\")\n    throwable.message = if (jsTypeOf(message) == \"undefined\") {\n        if (cause != null) cause.toString() else null\n    } else {\n        message\n    }\n    throwable.cause = cause\n    throwable.name = \"Throwable\"\n    return throwable\n}\n\n@JsName(\"BoxedChar\")\ninternal class BoxedChar(val c: Int) : Comparable<Int> {\n    override fun equals(other: Any?): Boolean {\n        return other is BoxedChar && c == other.c\n    }\n\n    override fun hashCode(): Int {\n        return c\n    }\n\n    override fun toString(): String {\n        return js(\"this.c\").unsafeCast<Char>().toString()\n    }\n\n    override fun compareTo(other: Int): Int {\n        return js(\"this.c - other\").unsafeCast<Int>()\n    }\n\n    @JsName(\"valueOf\")\n    public fun valueOf(): Int {\n        return c\n    }\n}\n\n@kotlin.internal.InlineOnly\ninternal inline fun <T> concat(args: Array<T>): T {\n    val typed = js(\"Array\")(args.size)\n    for (i in args.indices) {\n        val arr = args[i]\n        if (arr !is Array<*>) {\n            typed[i] = js(\"[]\").slice.call(arr)\n        } else {\n            typed[i] = arr\n        }\n    }\n    return js(\"[]\").concat.apply(js(\"[]\"), typed);\n}\n\n/** Concat regular Array's and TypedArray's into an Array.\n */\n@PublishedApi\n@JsName(\"arrayConcat\")\n@Suppress(\"UNUSED_PARAMETER\")\ninternal fun <T> arrayConcat(a: T, b: T): T {\n    return concat(js(\"arguments\"))\n}\n\n/** Concat primitive arrays. Main use: prepare vararg arguments.\n *  For compatibility with 1.1.0 the arguments may be a mixture of Array's and TypedArray's.\n *\n *  If the first argument is TypedArray (Byte-, Short-, Char-, Int-, Float-, and DoubleArray) returns a TypedArray, otherwise an Array.\n *  If the first argument has the $type$ property (Boolean-, Char-, and LongArray) copy its value to result.$type$.\n *  If the first argument is a regular Array without the $type$ property default to arrayConcat.\n */\n@PublishedApi\n@JsName(\"primitiveArrayConcat\")\n@Suppress(\"UNUSED_PARAMETER\")\ninternal fun <T> primitiveArrayConcat(a: T, b: T): T {\n    val args: Array<T> = js(\"arguments\")\n    if (a is Array<*> &&
```

a.asDynamic().`$type$` === undefined) {\n return concat(args)\n } else {\n var size = 0\n for (i in args.indices) {\n size += args[i].asDynamic().length as Int\n }\n val result = js(\"new a.constructor(size)\")\n kotlin.copyArrayType(a, result)\n size = 0\n for (i in args.indices) {\n val arr = args[i].asDynamic()\n for (j in 0 until arr.length) {\n result[size++] = arr[j]\n }\n }\n return result\n }\n}\n\n@JsName(\"booleanArrayOf\")\ninternal fun booleanArrayOf() = withType(\"BooleanArray\", js(\"[].slice.call(arguments)\"))\n\n@JsName(\"charArrayOf\") // The arguments have to be slice'd here because of Rhino (see KT-16974)\ninternal fun charArrayOf() = withType(\"CharArray\", js(\"new Uint16Array([].slice.call(arguments))\"))\n\n@JsName(\"longArrayOf\")\ninternal fun longArrayOf() = withType(\"LongArray\", js(\"[].slice.call(arguments)\"))\n\n@JsName(\"withType\")\n@kotlin.internal.InlineOnly\ninternal inline fun withType(type: String, array: dynamic): dynamic {\n array.`$type$` = type\n return array\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js\n\n/**\n * Function corresponding to JavaScript's `typeof` operator\n */\n@kotlin.internal.InlineOnly\n@Suppress(\"UNUSED_PARAMETER\")\npublic inline fun jsTypeOf(a: Any?): String = js(\"typeof a\")\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:Suppress(\"UNUSED_PARAMETER\", \"NOTHING_TO_INLINE\")\n\npackage kotlin\n\n/**\n * Returns an empty array of the specified type [T].\n */\npublic inline fun <T> emptyArray(): Array<T> = js(\"[]\")\n\n@library\npublic fun <T> arrayOf(vararg elements: T): Array<T> = definedExternally\n\n@library\npublic fun doubleArrayOf(vararg elements: Double): DoubleArray = definedExternally\n\n@library\npublic fun floatArrayOf(vararg elements: Float): FloatArray = definedExternally\n\n@library\npublic fun longArrayOf(vararg elements: Long): LongArray = definedExternally\n\n@library\npublic fun intArrayOf(vararg elements: Int): IntArray = definedExternally\n\n@library\npublic fun charArrayOf(vararg elements: Char): CharArray = definedExternally\n\n@library\npublic fun shortArrayOf(vararg elements: Short): ShortArray = definedExternally\n\n@library\npublic fun byteArrayOf(vararg elements: Byte): ByteArray = definedExternally\n\n@library\npublic fun booleanArrayOf(vararg elements: Boolean): BooleanArray = definedExternally\n\n/**\n * Creates a new instance of the [Lazy] that uses the specified initialization function [initializer].\n */\npublic actual fun <T> lazy(initializer: () -> T): Lazy<T> = UnsafeLazyImpl(initializer)\n\n/**\n * Creates a new instance of the [Lazy] that uses the specified initialization function [initializer].\n *\n * The [mode] parameter is ignored. */\npublic actual fun <T> lazy(mode: LazyThreadSafetyMode, initializer: () -> T): Lazy<T> = UnsafeLazyImpl(initializer)\n\n/**\n * Creates a new instance of the [Lazy] that uses the specified initialization function [initializer].\n *\n * The [lock] parameter is ignored.\n */\npublic actual fun <T> lazy(lock: Any?, initializer: () -> T): Lazy<T> = UnsafeLazyImpl(initializer)\n\n\ninternal fun fillFrom(src: dynamic, dst: dynamic): dynamic {\n val srcLen: Int = src.length\n val dstLen: Int = dst.length\n var index: Int = 0\n while (index < srcLen && index < dstLen) dst[index] = src[index++]\n return dst\n}\n\n\ninternal fun arrayCopyResize(source: dynamic, newSize: Int, defaultValue: Any?): dynamic {\n val result = source.slice(0, newSize)\n copyArrayType(source, result)\n var index: Int = source.length\n if (newSize > index) {\n result.length = newSize\n while (index < newSize) result[index++] = defaultValue\n }\n return result\n}\n\ninternal fun <T> arrayPlusCollection(array: dynamic, collection: Collection<T>): dynamic {\n val result = array.slice()\n result.length += collection.size\n copyArrayType(array, result)\n var index: Int = array.length\n for (element in collection) result[index++] = element\n return result\n}\n\ninternal fun <T> fillFromCollection(dst: dynamic, startIndex: Int, collection: Collection<T>): dynamic {\n var index = startIndex\n for (element in collection) dst[index++] = element\n return dst\n}\n\ninternal inline fun copyArrayType(from: dynamic, to: dynamic) {\n if (from.`$type$` !== undefined) {\n to.`$type$` = from.`$type$`\n }\n}\n\ninternal inline fun jsIsType(obj: dynamic, jsClass: dynamic) = js(\"Kotlin\").isType(obj, jsClass)","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that

can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\n/**\n * Creates a Char with the specified [code].\n *\n * @sample samples.text.Chars.charFromCode\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic actual inline fun Char(code: UShort): Char {\n   return code.toInt().toChar()\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.coroutines\n\nimport kotlin.coroutines.intrinsics.COROUTINE_SUSPENDED\n\n@SinceKotlin(\"1.3\")\n@JsName(\"CoroutineImpl\")\ninternal abstract class CoroutineImpl(private val resultContinuation: Continuation<Any?>) : Continuation<Any?> {\n   protected var state = 0\n   protected var exceptionState = 0\n   protected var result: Any? = null\n   protected var exception: Throwable? = null\n   protected var finallyPath: Array<Int>? = null\n\n   public override val context: CoroutineContext = resultContinuation.context\n\n   private var intercepted_: Continuation<Any?>? = null\n\n   public fun intercepted(): Continuation<Any?> =\n   intercepted_\n   ?: (context[ContinuationInterceptor]?.interceptContinuation(this) ?: this)\n   .also { intercepted_ = it }\n\n   override fun resumeWith(result: Result<Any?>) {\n   var current = this\n   var currentResult: Any? = result.getOrNull()\n   var currentException: Throwable? = result.exceptionOrNull()\n\n   // This loop unrolls recursion in current.resumeWith(param) to make saner and shorter stack traces on resume\n   while (true) {\n   with(current) {\n   val completion = resultContinuation\n\n   // Set result and exception fields in the current continuation\n   if (currentException == null) {\n   this.result = currentResult\n   } else {\n   state = exceptionState\n   exception = currentException\n   }\n\n   try {\n   val outcome = doResume()\n   if (outcome === COROUTINE_SUSPENDED) return\n   currentResult = outcome\n   currentException = null\n   } catch (exception: dynamic) { // Catch all exceptions\n   currentResult = null\n   currentException = exception.unsafeCast<Throwable>()\n   }\n\n   releaseIntercepted() // this state machine instance is terminating\n\n   if (completion is CoroutineImpl) {\n   // unrolling recursion via loop\n   current = completion\n   } else {\n   // top-level completion reached -- invoke and return\n   currentException?.let {\n   completion.resumeWithException(it)\n   } ?: completion.resume(currentResult)\n   return\n   }\n   }\n   }\n   }\n\n   private fun releaseIntercepted() {\n   val intercepted = intercepted_\n   if (intercepted != null && intercepted !== this) {\n   context[ContinuationInterceptor]!!.releaseInterceptedContinuation(intercepted)\n   }\n   this.intercepted_ = CompletedContinuation // just in case\n   }\n\n   protected abstract fun doResume(): Any?\n}\n\ninternal object CompletedContinuation : Continuation<Any?> {\n   override val context: CoroutineContext\n   get() = error(\"This continuation is already complete\")\n\n   override fun resumeWith(result: Result<Any?>) {\n   error(\"This continuation is already complete\")\n   }\n\n   override fun toString(): String = \"This continuation is already complete\"\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:Suppress(\"UNCHECKED_CAST\", \"RedundantVisibilityModifier\")\n\npackage kotlin\n\nimport kotlin.contracts.*\nimport kotlin.internal.InlineOnly\nimport kotlin.jvm.JvmField\nimport kotlin.jvm.JvmInline\nimport kotlin.jvm.JvmName\n\n/**\n * A discriminated union that encapsulates a successful outcome with a value of type [T]\n * or a failure with an arbitrary [Throwable] exception.\n */\n@SinceKotlin(\"1.3\")\n@JvmInline\npublic value class Result<out T> @PublishedApi internal constructor(\n   @PublishedApi\n   internal val value: Any?\n) : Serializable {\n   // discovery\n\n   /**\n   * Returns `true` if this instance represents a successful outcome.\n   * In this case [isFailure] returns `false`.\n   */\n   public val isSuccess: Boolean get() = value !is Failure\n\n   /**\n   * Returns `true` if this instance represents a failed outcome.\n   * In this case [isSuccess] returns `false`.\n   */\n   public val isFailure: Boolean get() = value is Failure\n\n   // value & exception retrieval\n\n   /**\n   * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or `null`\n   * if it is [failure][Result.isFailure].\n   *\n   * This function is a shorthand for `getOrElse { null }` (see [getOrElse]) or\n   * `fold(onSuccess = { it }, onFailure = { null })` (see [fold]).\n   */\n   @InlineOnly\n   public inline fun

```
getOrNull(): T? =\n        when {\n            isFailure -> null\n            else -> value as T\n        }\n\n    /**\n     * Returns the encapsulated [Throwable] exception if this instance represents [failure][isFailure] or `null`\n     * if it is [success][isSuccess].\n     *\n     * This function is a shorthand for `fold(onSuccess = { null }, onFailure = { it })` (see [fold]).\n     */\n    public fun exceptionOrNull(): Throwable? =\n        when (value) {\n            is Failure -> value.exception\n            else -> null\n        }\n\n    /**\n     * Returns a string `Success(v)` if this instance represents [success][Result.isSuccess]\n     * where `v` is a string representation of the value or a string `Failure(x)` if\n     * it is [failure][isFailure] where `x` is a string representation of the exception.\n     */\n    public override fun toString(): String =\n        when (value) {\n            is Failure -> value.toString() // \"Failure($exception)\"\n            else -> \"Success($value)\"\n        }\n\n    // companion with constructors\n\n    /**\n     * Companion object for [Result] class that contains its constructor functions\n     * [success] and [failure].\n     */\n    public companion object {\n        /**\n         * Returns an instance that encapsulates the given [value] as successful value.\n         */\n        @Suppress(\"INAPPLICABLE_JVM_NAME\")\n        @InlineOnly\n        @JvmName(\"success\")\n        public inline fun <T> success(value: T): Result<T> =\n            Result(value)\n\n        /**\n         * Returns an instance that encapsulates the given [Throwable] [exception] as failure.\n         */\n        @Suppress(\"INAPPLICABLE_JVM_NAME\")\n        @InlineOnly\n        @JvmName(\"failure\")\n        public inline fun <T> failure(exception: Throwable): Result<T> =\n            Result(createFailure(exception))\n    }\n\n    internal class Failure(\n        @JvmField\n        val exception: Throwable\n    ) : Serializable {\n        override fun equals(other: Any?): Boolean = other is Failure && exception == other.exception\n        override fun hashCode(): Int = exception.hashCode()\n        override fun toString(): String = \"Failure($exception)\"\n    }\n}\n\n/**\n * Creates an instance of internal marker [Result.Failure] class to\n * make sure that this class is not exposed in ABI.\n */\n@PublishedApi\n@SinceKotlin(\"1.3\")\ninternal fun createFailure(exception: Throwable): Any =\n    Result.Failure(exception)\n\n/**\n * Throws exception if the result is failure. This internal function minimizes\n * inlined bytecode for [getOrThrow] and makes sure that in the future we can\n * add some exception-augmenting logic here (if needed).\n */\n@PublishedApi\n@SinceKotlin(\"1.3\")\ninternal fun Result<*>.throwOnFailure() {\n    if (value is Result.Failure) throw value.exception\n}\n\n/**\n * Calls the specified function [block] and returns its encapsulated result if invocation was successful,\n * catching any [Throwable] exception that was thrown from the [block] function execution and encapsulating it as a failure.\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <R> runCatching(block: () -> R): Result<R> {\n    return try {\n        Result.success(block())\n    } catch (e: Throwable) {\n        Result.failure(e)\n    }\n}\n\n/**\n * Calls the specified function [block] with `this` value as its receiver and returns its encapsulated result if invocation was successful,\n * catching any [Throwable] exception that was thrown from the [block] function execution and encapsulating it as a failure.\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <T, R> T.runCatching(block: T.() -> R): Result<R> {\n    return try {\n        Result.success(block())\n    } catch (e: Throwable) {\n        Result.failure(e)\n    }\n}\n\n// -- extensions ---\n\n/**\n * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or throws the encapsulated [Throwable] exception\n * if it is [failure][Result.isFailure].\n *\n * This function is a shorthand for `getOrElse { throw it }` (see [getOrElse]).\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <T> Result<T>.getOrThrow(): T {\n    throwOnFailure()\n    return value as T\n}\n\n/**\n * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or the\n * result of [onFailure] function for the encapsulated [Throwable] exception if it is [failure][Result.isFailure].\n *\n * Note, that this function rethrows any [Throwable] exception thrown by [onFailure] function.\n *\n * This function is a shorthand for `fold(onSuccess = { it }, onFailure = onFailure)` (see [fold]).\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <R, T : R> Result<T>.getOrElse(onFailure: (exception: Throwable) -> R): R {\n    contract {\n        callsInPlace(onFailure, InvocationKind.AT_MOST_ONCE)\n    }\n    return when (val exception = exceptionOrNull()) {\n        null -> value as T\n        else -> onFailure(exception)\n    }\n}\n\n/**\n * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or the\n * [defaultValue] if it is [failure][Result.isFailure].\n *\n * This function is a shorthand for `getOrElse { defaultValue }` (see [getOrElse]).\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <R, T : R> Result<T>.getOrDefault(defaultValue: R): R {\n    if (isFailure) return defaultValue\n    return value as T\n}\n\n/**\n * Returns the result of [onSuccess] for the
```

encapsulated value if this instance represents [success][Result.isSuccess]\n * or the result of [onFailure] function for the encapsulated [Throwable] exception if it is [failure][Result.isFailure].\n *\n * Note, that this function rethrows any [Throwable] exception thrown by [onSuccess] or by [onFailure] function.\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <R, T> Result<T>.fold(\n onSuccess: (value: T) -> R,\n onFailure: (exception: Throwable) -> R\n): R {\n contract {\n callsInPlace(onSuccess, InvocationKind.AT_MOST_ONCE)\n callsInPlace(onFailure, InvocationKind.AT_MOST_ONCE)\n }\n return when (val exception = exceptionOrNull()) {\n null -> onSuccess(value as T)\n else -> onFailure(exception)\n }\n}\n\n// transformation\n\n/**\n * Returns the encapsulated result of the given [transform] function applied to the encapsulated value\n * if this instance represents [success][Result.isSuccess] or the\n * original encapsulated [Throwable] exception if it is [failure][Result.isFailure].\n *\n * Note, that this function rethrows any [Throwable] exception thrown by [transform] function.\n * See [mapCatching] for an alternative that encapsulates exceptions.\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <R, T> Result<T>.map(transform: (value: T) -> R): Result<R> {\n contract {\n callsInPlace(transform, InvocationKind.AT_MOST_ONCE)\n }\n return when {\n isSuccess -> Result.success(transform(value as T))\n else -> Result(value)\n }\n}\n\n/**\n * Returns the encapsulated result of the given [transform] function applied to the encapsulated value\n * if this instance represents [success][Result.isSuccess] or the\n * original encapsulated [Throwable] exception if it is [failure][Result.isFailure].\n *\n * This function catches any [Throwable] exception thrown by [transform] function and encapsulates it as a failure.\n * See [map] for an alternative that rethrows exceptions from `transform` function.\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <R, T> Result<T>.mapCatching(transform: (value: T) -> R): Result<R> {\n return when {\n isSuccess -> runCatching { transform(value as T) }\n else -> Result(value)\n }\n}\n\n/**\n * Returns the encapsulated result of the given [transform] function applied to the encapsulated [Throwable] exception\n * if this instance represents [failure][Result.isFailure] or the\n * original encapsulated value if it is [success][Result.isSuccess].\n *\n * Note, that this function rethrows any [Throwable] exception thrown by [transform] function.\n * See [recoverCatching] for an alternative that encapsulates exceptions.\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <R, T : R> Result<T>.recover(transform: (exception: Throwable) -> R): Result<R> {\n contract {\n callsInPlace(transform, InvocationKind.AT_MOST_ONCE)\n }\n return when (val exception = exceptionOrNull()) {\n null -> this\n else -> Result.success(transform(exception))\n }\n}\n\n/**\n * Returns the encapsulated result of the given [transform] function applied to the encapsulated [Throwable] exception\n * if this instance represents [failure][Result.isFailure] or the\n * original encapsulated value if it is [success][Result.isSuccess].\n *\n * This function catches any [Throwable] exception thrown by [transform] function and encapsulates it as a failure.\n * See [recover] for an alternative that rethrows exceptions.\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <R, T : R> Result<T>.recoverCatching(transform: (exception: Throwable) -> R): Result<R> {\n return when (val exception = exceptionOrNull()) {\n null -> this\n else -> runCatching { transform(exception) }\n }\n}\n\n// \"peek\" onto value/exception and pipe\n\n/**\n * Performs the given [action] on the encapsulated [Throwable] exception if this instance represents [failure][Result.isFailure].\n * Returns the original `Result` unchanged.\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <T> Result<T>.onFailure(action: (exception: Throwable) -> Unit): Result<T> {\n contract {\n callsInPlace(action, InvocationKind.AT_MOST_ONCE)\n }\n exceptionOrNull()?.let { action(it) }\n return this\n}\n\n/**\n * Performs the given [action] on the encapsulated value if this instance represents [success][Result.isSuccess].\n * Returns the original `Result` unchanged.\n */\n@InlineOnly\n@SinceKotlin(\"1.3\")\npublic inline fun <T> Result<T>.onSuccess(action: (value: T) -> Unit): Result<T> {\n contract {\n callsInPlace(action, InvocationKind.AT_MOST_ONCE)\n }\n if (isSuccess) action(value as T)\n return this\n}\n\n// -------------------\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.coroutines\n\nimport kotlin.contracts.*\nimport kotlin.coroutines.intrinsics.*\nimport kotlin.internal.InlineOnly\n\n/**\n * Interface representing a continuation after a suspension point that returns a value of type `T`.\n

```
*/\n@SinceKotlin(\"1.3\")\npublic interface Continuation<in T> {\n    /**\n     * The context of the coroutine that corresponds to this continuation.\n     */\n    public val context: CoroutineContext\n\n    /**\n     * Resumes the execution of the corresponding coroutine passing a successful or failed [result] as the\n     * return value of the last suspension point.\n     */\n    public fun resumeWith(result: Result<T>)\n}\n\n/**\n * Classes and interfaces marked with this annotation are restricted when used as receivers for extension\n * `suspend` functions. These `suspend` extensions can only invoke other member or extension `suspend` functions on this particular\n * receiver and are restricted from calling arbitrary suspension functions.\n */\n@SinceKotlin(\"1.3\")\n@Target(AnnotationTarget.CLASS)\n@Retention(AnnotationRetention.BINARY)\npublic annotation class RestrictsSuspension\n\n/**\n * Resumes the execution of the corresponding coroutine passing [value] as the return value of the last suspension point.\n */\n@SinceKotlin(\"1.3\")\n@InlineOnly\npublic inline fun <T> Continuation<T>.resume(value: T): Unit =\n    resumeWith(Result.success(value))\n\n/**\n * Resumes the execution of the corresponding coroutine so that the [exception] is re-thrown right after the\n * last suspension point.\n */\n@SinceKotlin(\"1.3\")\n@InlineOnly\npublic inline fun <T> Continuation<T>.resumeWithException(exception: Throwable): Unit =\n    resumeWith(Result.failure(exception))\n\n/**\n * Creates a [Continuation] instance with the given [context] and implementation of [resumeWith] method.\n */\n@SinceKotlin(\"1.3\")\n@InlineOnly\npublic inline fun <T> Continuation(\n    context: CoroutineContext,\n    crossinline resumeWith: (Result<T>) -> Unit\n): Continuation<T> =\n    object : Continuation<T> {\n        override val context: CoroutineContext\n            get() = context\n\n        override fun resumeWith(result: Result<T>) =\n            resumeWith(result)\n    }\n\n/**\n * Creates a coroutine without a receiver and with result type [T].\n * This function creates a new, fresh instance of suspendable computation every time it is invoked.\n *\n * To start executing the created coroutine, invoke `resume(Unit)` on the returned [Continuation] instance.\n * The [completion] continuation is invoked when the coroutine completes with a result or an exception.\n * Subsequent invocation of any resume function on the resulting continuation will produce an [IllegalStateException].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"UNCHECKED_CAST\")\npublic fun <T> (suspend () -> T).createCoroutine(\n    completion: Continuation<T>\n): Continuation<Unit> =\n    SafeContinuation(createCoroutineUnintercepted(completion).intercepted(), COROUTINE_SUSPENDED)\n\n/**\n * Creates a coroutine with receiver type [R] and result type [T].\n * This function creates a new, fresh instance of suspendable computation every time it is invoked.\n *\n * To start executing the created coroutine, invoke `resume(Unit)` on the returned [Continuation] instance.\n * The [completion] continuation is invoked when the coroutine completes with a result or an exception.\n * Subsequent invocation of any resume function on the resulting continuation will produce an [IllegalStateException].\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"UNCHECKED_CAST\")\npublic fun <R, T> (suspend R.() -> T).createCoroutine(\n    receiver: R,\n    completion: Continuation<T>\n): Continuation<Unit> =\n    SafeContinuation(createCoroutineUnintercepted(receiver, completion).intercepted(), COROUTINE_SUSPENDED)\n\n/**\n * Starts a coroutine without a receiver and with result type [T].\n * This function creates and starts a new, fresh instance of suspendable computation every time it is invoked.\n * The [completion] continuation is invoked when the coroutine completes with a result or an exception.\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"UNCHECKED_CAST\")\npublic fun <T> (suspend () -> T).startCoroutine(\n    completion: Continuation<T>\n) {\n    createCoroutineUnintercepted(completion).intercepted().resume(Unit)\n}\n\n/**\n * Starts a coroutine with receiver type [R] and result type [T].\n * This function creates and starts a new, fresh instance of suspendable computation every time it is invoked.\n * The [completion] continuation is invoked when the coroutine completes with a result or an exception.\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"UNCHECKED_CAST\")\npublic fun <R, T> (suspend R.() -> T).startCoroutine(\n    receiver: R,\n    completion: Continuation<T>\n) {\n    createCoroutineUnintercepted(receiver, completion).intercepted().resume(Unit)\n}\n\n/**\n * Obtains the current continuation instance inside suspend functions and suspends\n * the currently running coroutine.\n *\n * In this function both [Continuation.resume] and [Continuation.resumeWithException] can be used either synchronously in\n * the same stack-frame where the suspension function is run or asynchronously later in the same thread or\n *
```

from a different thread of execution. Subsequent invocation of any resume function will produce an [IllegalStateException].\n */\n@SinceKotlin(\"1.3\")\n@InlineOnly\npublic suspend inline fun <T> suspendCoroutine(crossinline block: (Continuation<T>) -> Unit): T {\n contract { callsInPlace(block, InvocationKind.EXACTLY_ONCE) }\n return suspendCoroutineUninterceptedOrReturn { c: Continuation<T> ->\n val safe = SafeContinuation(c.intercepted())\n block(safe)\n safe.getOrThrow()\n }\n}\n\n/**\n * Returns the context of the current coroutine.\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"WRONG_MODIFIER_TARGET\")\n@InlineOnly\npublic suspend inline val coroutineContext: CoroutineContext\n get() {\n throw NotImplementedError(\"Implemented as intrinsic\")\n }\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.coroutines.intrinsics\n\nimport kotlin.coroutines.*\nimport kotlin.internal.InlineOnly\n\n/**\n * Starts an unintercepted coroutine without a receiver and with result type [T] and executes it until its first suspension.\n * Returns the result of the coroutine or throws its exception if it does not suspend or [COROUTINE_SUSPENDED] if it suspends.\n * In the latter case, the [completion] continuation is invoked when the coroutine completes with a result or an exception.\n *\n * The coroutine is started directly in the invoker's thread without going through the [ContinuationInterceptor] that might\n * be present in the completion's [CoroutineContext]. It is the invoker's responsibility to ensure that a proper invocation\n * context is established.\n *\n * This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended\n * coroutine using a reference to the suspending function.\n */\n@SinceKotlin(\"1.3\")\n@InlineOnly\npublic actual inline fun <T> (suspend () -> T).startCoroutineUninterceptedOrReturn(\n completion: Continuation<T>\n): Any? = this.asDynamic()(completion, false)\n\n/**\n * Starts an unintercepted coroutine with receiver type [R] and result type [T] and executes it until its first suspension.\n * Returns the result of the coroutine or throws its exception if it does not suspend or [COROUTINE_SUSPENDED] if it suspends.\n * In the latter case, the [completion] continuation is invoked when the coroutine completes with a result or an exception.\n *\n * The coroutine is started directly in the invoker's thread without going through the [ContinuationInterceptor] that might\n * be present in the completion's [CoroutineContext]. It is the invoker's responsibility to ensure that a proper invocation\n * context is established.\n *\n * This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended\n * coroutine using a reference to the suspending function.\n */\n@SinceKotlin(\"1.3\")\n@InlineOnly\npublic actual inline fun <R, T> (suspend R.() -> T).startCoroutineUninterceptedOrReturn(\n receiver: R,\n completion: Continuation<T>\n): Any? = this.asDynamic()(receiver, completion, false)\n\n@InlineOnly\ninternal actual inline fun <R, P, T> (suspend R.(P) -> T).startCoroutineUninterceptedOrReturn(\n receiver: R,\n param: P,\n completion: Continuation<T>\n): Any? = this.asDynamic()(receiver, param, completion, false)\n\n/**\n * Creates unintercepted coroutine without receiver and with result type [T].\n * This function creates a new, fresh instance of suspendable computation every time it is invoked.\n *\n * To start executing the created coroutine, invoke `resume(Unit)` on the returned [Continuation] instance.\n * The [completion] continuation is invoked when coroutine completes with result or exception.\n *\n * This function returns unintercepted continuation.\n * Invocation of `resume(Unit)` starts coroutine immediately in the invoker's call stack without going through the\n * [ContinuationInterceptor] that might be present in the completion's [CoroutineContext].\n * It is the invoker's responsibility to ensure that a proper invocation context is established.\n * Note that [completion] of this function may get invoked in an arbitrary context.\n *\n * [Continuation.intercepted] can be used to acquire the intercepted continuation.\n * Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of\n * both the coroutine and [completion] happens in the invocation context established by\n * [ContinuationInterceptor].\n *\n * Repeated invocation of any resume function on the resulting continuation corrupts the\n * state machine of the coroutine and may result in arbitrary behaviour or exception.\n */\n@SinceKotlin(\"1.3\")\npublic actual fun <T> (suspend () -> T).createCoroutineUnintercepted(\n completion: Continuation<T>\n): Continuation<Unit> =\n // Kotlin/JS suspend lambdas have an extra parameter `suspended`\n if (this.asDynamic().length == 2) {\n // When `suspended` is true the continuation is created,

but not executed\n this.asDynamic()(completion, true)\n } else {\n createCoroutineFromSuspendFunction(completion) {\n this.asDynamic()(completion)\n }\n }\n\n/**\n * Creates unintercepted coroutine with receiver type [R] and result type [T].\n * This function creates a new, fresh instance of suspendable computation every time it is invoked.\n *\n * To start executing the created coroutine, invoke `resume(Unit)` on the returned [Continuation] instance.\n * The [completion] continuation is invoked when coroutine completes with result or exception.\n *\n * This function returns unintercepted continuation.\n * Invocation of `resume(Unit)` starts coroutine immediately in the invoker's call stack without going through the\n * [ContinuationInterceptor] that might be present in the completion's [CoroutineContext].\n * It is the invoker's responsibility to ensure that a proper invocation context is established.\n * Note that [completion] of this function may get invoked in an arbitrary context.\n *\n * [Continuation.intercepted] can be used to acquire the intercepted continuation.\n * Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of\n * both the coroutine and [completion] happens in the invocation context established by\n * [ContinuationInterceptor].\n *\n * Repeated invocation of any resume function on the resulting continuation corrupts the\n * state machine of the coroutine and may result in arbitrary behaviour or exception.\n */\n@SinceKotlin(\"1.3\")\npublic actual fun <R, T> (suspend R.() -> T).createCoroutineUnintercepted(\n receiver: R,\n completion: Continuation<T>\n): Continuation<Unit> =\n // Kotlin/JS suspend lambdas have an extra parameter `suspended`\n if (this.asDynamic().length == 3) {\n // When `suspended` is true the continuation is created, but not executed\n this.asDynamic()(receiver, completion, true)\n } else {\n createCoroutineFromSuspendFunction(completion) {\n this.asDynamic()(receiver, completion)\n }\n }\n\n/**\n * Intercepts this continuation with [ContinuationInterceptor].\n *\n * This function shall be used on the immediate result of [createCoroutineUnintercepted] or [suspendCoroutineUninterceptedOrReturn],\n * in which case it checks for [ContinuationInterceptor] in the continuation's [context][Continuation.context],\n * invokes [ContinuationInterceptor.interceptContinuation], caches and returns the result.\n *\n * If this function is invoked on other [Continuation] instances it returns `this` continuation unchanged.\n */\n@SinceKotlin(\"1.3\")\npublic actual fun <T> Continuation<T>.intercepted(): Continuation<T> =\n (this as? CoroutineImpl)?.intercepted() ?: this\n\n\nprivate inline fun <T> createCoroutineFromSuspendFunction(\n completion: Continuation<T>,\n crossinline block: () -> Any?\n): Continuation<Unit> {\n @Suppress(\"UNCHECKED_CAST\")\n return object : CoroutineImpl(completion as Continuation<Any?>) {\n override fun doResume(): Any? {\n exception?.let { throw it }\n return block()\n }\n }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js\n\n// Mirrors signature from JS IR BE\n// Used for js.translator/testData/box/number/mulInt32.kt\n@library\n@JsName(\"imulEmulated\")\n@Suppress(\"UNUSED_PARAMETER\")\ninternal fun imul(x: Int, y: Int): Int = definedExternally\n\n@Suppress(\"NOTHING_TO_INLINE\")\ninternal inline fun isArrayish(o: dynamic) = js(\"Kotlin\").isArrayish(o)\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\n// NOTE: Do not author your exceptions as they are written in this file, instead use this template:\n/*\npublic open class MyException : Exception {\n constructor() : super()\n constructor(message: String?) : super(message)\n constructor(message: String?, cause: Throwable?) : super(message, cause)\n constructor(cause: Throwable?) : super(cause)\n}\n*/\n\n\n// TODO: remove primary constructors, make all secondary KT-22055\n\n@Suppress(\"USELESS_ELVIS_RIGHT_IS_NULL\")\npublic actual open class Error actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null) {\n actual constructor() : this(null, null)\n actual constructor(message: String?) : this(message, null)\n actual constructor(cause: Throwable?) : this(undefined, cause)\n}\n\n@Suppress(\"USELESS_ELVIS_RIGHT_IS_NULL\")\npublic actual open class Exception actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null) {\n actual constructor() : this(null, null)\n actual constructor(message: String?) : this(message, null)\n actual constructor(cause:

Throwable?) : this(undefined, cause)\n}\n\npublic actual open class RuntimeException actual constructor(message: String?, cause: Throwable?) : Exception(message, cause) {\n   actual constructor() : this(null, null)\n   actual constructor(message: String?) : this(message, null)\n   actual constructor(cause: Throwable?) : this(undefined, cause)\n}\n\npublic actual open class IllegalArgumentException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {\n   actual constructor() : this(null, null)\n   actual constructor(message: String?) : this(message, null)\n   actual constructor(cause: Throwable?) : this(undefined, cause)\n}\n\npublic actual open class IllegalStateException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {\n   actual constructor() : this(null, null)\n   actual constructor(message: String?) : this(message, null)\n   actual constructor(cause: Throwable?) : this(undefined, cause)\n}\n\npublic actual open class IndexOutOfBoundsException actual constructor(message: String?) : RuntimeException(message) {\n   actual constructor() : this(null)\n}\n\npublic actual open class ConcurrentModificationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {\n   actual constructor() : this(null, null)\n   actual constructor(message: String?) : this(message, null)\n   actual constructor(cause: Throwable?) : this(undefined, cause)\n}\n\npublic actual open class UnsupportedOperationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {\n   actual constructor() : this(null, null)\n   actual constructor(message: String?) : this(message, null)\n   actual constructor(cause: Throwable?) : this(undefined, cause)\n}\n\n\npublic actual open class NumberFormatException actual constructor(message: String?) : IllegalArgumentException(message) {\n   actual constructor() : this(null)\n}\n\n\npublic actual open class NullPointerException actual constructor(message: String?) : RuntimeException(message) {\n   actual constructor() : this(null)\n}\n\npublic actual open class ClassCastException actual constructor(message: String?) : RuntimeException(message) {\n   actual constructor() : this(null)\n}\n\npublic actual open class AssertionError\n@SinceKotlin(\"1.4\")\nconstructor(message: String?, cause: Throwable?) : Error(message, cause) {\n   actual constructor() : this(null)\n   constructor(message: String?) : this(message, null)\n   actual constructor(message: Any?) : this(message.toString(), message as? Throwable)\n}\n\npublic actual open class NoSuchElementException actual constructor(message: String?) : RuntimeException(message) {\n   actual constructor() : this(null)\n}\n\n@SinceKotlin(\"1.3\")\npublic actual open class ArithmeticException actual constructor(message: String?) : RuntimeException(message) {\n   actual constructor() : this(null)\n}\n\npublic actual open class NoWhenBranchMatchedException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {\n   actual constructor() : this(null, null)\n   actual constructor(message: String?) : this(message, null)\n   actual constructor(cause: Throwable?) : this(undefined, cause)\n}\n\npublic actual open class UninitializedPropertyAccessException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {\n   actual constructor() : this(null, null)\n   actual constructor(message: String?) : this(message, null)\n   actual constructor(cause: Throwable?) : this(undefined, cause)\n}\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. Use of this source code is governed by the Apache 2.0 license\n * that can be found in the license/LICENSE.txt file.\n */\n\n\n@file:Suppress(\"UNUSED_PARAMETER\")\n\npackage kotlin.js\n\n@kotlin.internal.InlineOnly\ninternal inline fun jsDeleteProperty(obj: Any, property: Any) {\n   js(\"delete obj[property]\")\n}\n\n@kotlin.internal.InlineOnly\ninternal inline fun jsBitwiseOr(lhs: Any?, rhs: Any?): Int =\n   js(\"lhs | rhs\").unsafeCast<Int>()","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\npackage kotlin.math\n\n/**\n * Returns this value with the sign bit same as of the [sign] value.\n *\n * If [sign] is `NaN` the sign of the result is undefined.\n */\n@SinceKotlin(\"1.2\")\npublic actual fun Double.withSign(sign: Double): Double {\n   val thisSignBit = js(\"Kotlin\").doubleSignBit(this).unsafeCast<Int>()\n   val newSignBit = js(\"Kotlin\").doubleSignBit(sign).unsafeCast<Int>()\n   return if (thisSignBit == newSignBit) this else -this\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\n/**\n * Returns a bit representation of the specified floating-point value as [Long]\n *

according to the IEEE 754 floating-point \"double format\" bit layout.\n
*/\n@SinceKotlin(\"1.2\")\n@library(\"doubleToBits\")\npublic actual fun Double.toBits(): Long =
definedExternally\n\n/**\n * Returns a bit representation of the specified floating-point value as [Long]\n *
according to the IEEE 754 floating-point \"double format\" bit layout,\n * preserving `NaN` values exact layout.\n
*/\n@SinceKotlin(\"1.2\")\n@library(\"doubleToRawBits\")\npublic actual fun Double.toRawBits(): Long =
definedExternally\n\n/**\n * Returns the [Double] value corresponding to a given bit representation.\n
*/\n@SinceKotlin(\"1.2\")\n@kotlin.internal.InlineOnly\npublic actual inline fun Double.Companion.fromBits(bits:
Long): Double = js(\"Kotlin\").doubleFromBits(bits).unsafeCast<Double>()\n\n/**\n * Returns a bit representation
of the specified floating-point value as [Int]\n * according to the IEEE 754 floating-point \"single format\" bit
layout.\n *\n * Note that in Kotlin/JS [Float] range is wider than \"single format\" bit layout can represent,\n * so
some [Float] values may overflow, underflow or loose their accuracy after conversion to bits and back.\n
*/\n@SinceKotlin(\"1.2\")\n@library(\"floatToBits\")\npublic actual fun Float.toBits(): Int =
definedExternally\n\n/**\n * Returns a bit representation of the specified floating-point value as [Int]\n * according
to the IEEE 754 floating-point \"single format\" bit layout,\n * preserving `NaN` values exact layout.\n *\n * Note
that in Kotlin/JS [Float] range is wider than \"single format\" bit layout can represent,\n * so some [Float] values
may overflow, underflow or loose their accuracy after conversion to bits and back.\n
*/\n@SinceKotlin(\"1.2\")\n@library(\"floatToRawBits\")\npublic actual fun Float.toRawBits(): Int =
definedExternally\n\n/**\n * Returns the [Float] value corresponding to a given bit representation.\n
*/\n@SinceKotlin(\"1.2\")\n@kotlin.internal.InlineOnly\npublic actual inline fun Float.Companion.fromBits(bits:
Int): Float =
js(\"Kotlin\").floatFromBits(bits).unsafeCast<Float>()\n\n\n@Suppress(\"NOTHING_TO_INLINE\")\ninternal
inline fun Long(low: Int, high: Int) = js(\"Kotlin\").Long.fromBits(low, high).unsafeCast<Long>()\ninternal inline
val Long.low: Int get() = this.asDynamic().getLowBits().unsafeCast<Int>()\ninternal inline val Long.high: Int get()
= this.asDynamic().getHighBits().unsafeCast<Int>()\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin
Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\n */\n\nimport kotlin.reflect.KClass\n\n@PublishedApi\ninternal fun <T :
Annotation> KClass<*>.findAssociatedObject(@Suppress(\"UNUSED_PARAMETER\") annotationClass:
KClass<T>): Any? {\n   // This API is not supported in js-v1. Return `null` to be source-compatible with js-ir.\n
return null\n}\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n *
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*/\n\npackage kotlin.text\n\n/**\n * Returns a string representation of this [Long] value in the specified [radix].\n
*\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n
*/\n@SinceKotlin(\"1.2\")\npublic actual fun Long.toString(radix: Int): String =
asDynamic().toString(checkRadix(radix))","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming
Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the
GenerateUnicodeData.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\n// 1343 ranges
totally\nprivate object Category {\n   val decodedRangeStart: IntArray\n   val decodedRangeCategory: IntArray\n
\n   init {\n   val toBase64 =
\"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/\"\n   val fromBase64 =
IntArray(128)\n   for (i in toBase64.indices) {\n   fromBase64[toBase64[i].code] = i\n   }\n   \n   //
rangeStartDiff.length = 1482\n   val rangeStartDiff =
\"gBCFEDCKCDCaDDaDBhBCEEDDDDDEDXBHYBH5BRwBGDCHDCIDFHDCHFDCDEIRTEE7BGHDDJl
CBbSEMOFGERwDEDDDDECEFCRBJhBFDCYFFCCzBvBjBBFC3BOhDBmBDGpBDDCtBBJIbEECLGDFC
LDCgBBKVKEDiDDHCFECECKCEODBebC5CLBOKhBJDDDDWEBHFCFCPBZDEL1BVBSLPBgBB2BDB
DICFBHKCCKCPDBHEDWBHEDDDDEDEDIBDGDCKCCGDDDCGECCWBFMDDCDEDDCHDDHKDDBK
DBHFCWBFGFDBDDFEDBPDDKCHBGDCHEDWBFGFDCEDEDBHDDGDCKCGJEGDBFDDFDDDDDME
FDBFDCGBOKDFDFDCGFCXBQDDDDDBEGEDFDDKHBHDDGFCXBKBFCEFCFCHCHECCKDNCCHFC

oBEDECFDDDDHDCCKJBGDCSDYBJEHBFDDEBIGKDCMuBFHEBGBIBKCkBFBFBXEIFJDFDGCKCEgB
BDPEDGKKGECIBkBEOBDFFLBkBBIBEFFEClBrBCEBEGDBKGGDDDDDCHDENDCFEKDDlBDDFrBCD
pKBECGEECpBBEChBBECGEECPB5BBECjCCDJUDQKG2CCGDsTCRBaCDrCDDIHNBEDLSDCJSCMLFC
CM0BDHGFLBFDDKGKGEFDDBKGjBB1BHFChBDFmCKfDDDDDDCGDCFDKeCFLsBEaGKBDiBXDDD1
BDGDEIGJEKGKGHBGCMF/BEBvBCEDDFHEKHKJJDDeDDGDKsBFEDCIEkBIICCDFKDDKeGCJHrBCDI
IDBNBHEBEFDBFsB/BNBiBlB6BBF1EIiDJIGCGCIIIIGCGCIIIIOCIIIIIIDFEDDBFEDDDDEBDIFDDFEDBLF
GCEEICFBJCDEDCLDKBFBKCCGDDKDDNDgBQNEBDMPFFDEDEBFFHECEBEEDFBEDDQjBCEDEFFC
CJHBeEEfsIIEUCHCxCBeZoBGlCZLV8BuCW3FBJB2BIvDB4HOesBFCfKQgIjEW/BEgBCiIwBVCGnBCgBBp
DvBBuBEDBHEFGCCjDCGEDCFCFlBDDF4BHCOBXJHBHBHBHBHBHBHBHBgBCECGHGEDIFBKCEDM
EtBaB5CM2GaMEDDCKCGFCJEDFDDDC2CDDDB6CDCFrBB+CDEKgBkBMQfBKeIBPgBKnBPgKguGgC9
vUDVB3jBD3BJoBGCsIBDQKCUuBDDKCcCCmCKCGIXJCNC/BBHGKDECEVFBEMCEEBqBDDGDFDXD
CEBDGEG0BEICyBQCICKGSGDEBKcICXLCLBdDDBvBDECCDNCKECFCJKFBpBFEDCJDBICCKCEQBG
DDByBEDCEFBYDCLEDDCKGCGCGJHBHBrBBEJDEwCjBIDCKGk9KMXExBEggCgoGuLCqDmBHMFFC
KBNBFBIsDQRrLCQgCC2BoBMCCQGEGQDCQDDDDFDGDECEEFBnEEBFEDCKCDCaDDaDBFCKBtBCf
DGCGCFEDDDDCECKDC\"\n val diff = decodeVarLenBase64(rangeStartDiff, fromBase64, 1342)\n val
start = IntArray(diff.size + 1)\n for (i in diff.indices) {\n start[i + 1] = start[i] + diff[i]\n }\n
decodedRangeStart = start\n \n // rangeCategory.length = 2033\n val rangeCategory =
\"PsY44a41W54UYJYZYB14W7XC15WZPsYa84bl9Zw8b85Lr7C44brlerrYBZBCZCiBiBiBhCiiBhChiBhiCBhh
ChiCihBhChCChiBhChiClBCFhjCiBiBihDhiBhCCihBiBBhCCFCEbEbEb7EbGhCk7BixRkiCi4BRbh4BhRhCBR
BCiiBBCiBChiZBCBCiBcGHhChCiBRBxxEYC40Rx8c6RGUm4GRFRFYRQZ44acG4wRYFEFGJYllGFlYGwc
GmkEmcGFJFl8cYxwFGFGRFGFRJFGkkcYkxRm6aFGEGmmEmEGRYRFGxxYFRFRFRGQGIFmIFIGIooGF
GFGYJ4EFmoIRFlxRlxRFRFxlRxlFllRxmFIGxxIoxRomFRIRxlFlmGRJFaL86F4mRxmGoRFRFRFRFllRxGIGR
xmGxmGmxRxGRFlRRJmmFllGYRmmIRFllRlRFRFllRFxxGFIGmmRoxImxRFRllGmxRJ4aRFGxmIoRFlxRlxR
FRFllRFxxGlImoGmmRxoIxoIGRmmIRxlFlmGRJ8FLRxmFFRFllRllRxxFlRlxRxlFRFRFRooGRIooRomRxFRIR
JLc8aRmoIoGFllRlRFRFRlmGmoIooRGRGRxmGFRllGmxRJRYL8lGooYFllRlRFRFRFRmlIIxGooRGRIRlxFG
RJxlFRGIFllRlRFlmGIGxIooRomF8xRxxFllILFGRJLcFxmIoRFRFRFxlRFRxxGxxIooGmmRRIRJxxIoYRFllGG
RaFEGYJYRxlFRFRFlRFllGGlxRFxEGRJRFRFcY84c8mGcJL8G1WIFRFRGIGmmYFGRGRcGc88RYcYRFIGI
GmmIomGFJYFooGmlFllGmmFIFIFGFmoIGIomFJIm8cBhRRxxBC4ECFRFRFlRFRFRFRFRFRFRFlRFRFRFRFR
FRGYLRFcRBRCxxUF8YFMF1WRFYKFRFRFGRFGYRFGRFllRlRGRFmmIGIooGGY44E46FmxRJRLRY44
U44GmmQRJRFEFRFGFlGRFRFxmGmoIooGmoIoxRxxIoGIGRxxcx4YJFRFRFRFRJLRcFmmIomRx4YFoGG
mRomIGIGmxRJRJRYEYRGmmHRGIFmIGmIIooGFRJYcGcRmmIFomGmmIomGmlFJFmoGooGGIRYFIGIG
RYJRFJFEYCRBRBYRGYGIGFGFllGomGFRCECECEGRGhCCiBCBCRBRCBCBCRBRCxBCBCRCDCDCD
CiiRBj7CbCiiRBj7b7iCiiRxiCBRbCBbxxCiiRBj7bRMQUY9+V9+VYtOQMY9eY43X44Z1WY54XYMQRQrER
LZ12ELZ12RERaRGHGHGR88B88BihBhiChhC8hcZBc8BB8CBCFi8cihBZBC8Z8CLKhCKr8cRZcZc88ZcZc85
Z8ZcZc1WcZc1WcZcZcZcRcRLcLcZcZcZcZc1WLcZ1WZ1WZcZ1WZ1WZ1WZcZcZcRcRcBRCixBBCiBBihC
CEBhCCchCGhCRY44LCiRRxxCFRkYRGFRFRFRFRFRFRFRFRFRFRGY9eY49eY44U49e49e1WYEYUY04VY
48cRcRcRcRcRs4Y48ElK1Wc1W12U2cKGooUE88KqqEl4c8RFxxGm7bkkFUF4kEkFRFRFx8cLcFcRFcRLcLc
LcLcLcFcFRFEFRcRFEYFEYFJFRhClmHnnYG4EhCEGFKGYRbEbhCCiBECiBhCk7bhClBihCiBBCBhCRhiBh
hCCRhiFkkCFlGllGllGFooGmIcGRL88aRFYRIFIGRYJRGFYl4FGJFGYFGIRYFRGIFmoIGIGIYxEJRYFmEFJ
FRFGmoImoIGRFGFmIRJRYFEFcloGIFmlGmlFGFlmGFRllEYFomGo4YlkEoGRFRFRFRFRFRCbECk7bRCFo
oG4oGRJRFRFRFRTSFRFRCRCRlGFZFRFRlxFFbRF2VRFRFRF6cRGY41WRG40UX1W44V24Y44X33Y44R
44U1WY50Z5R46YRFRFxxQY44a41W54UYJYZYB14W7XC15WZ12YYFEFEFRFRFRFlxRllRxxa65b86axcZc
RQcR\"\n decodedRangeCategory = decodeVarLenBase64(rangeCategory, fromBase64, 1343)\n
}\n}\n\nprivate fun categoryValueFrom(code: Int, ch: Int): Int {\n return when {\n code < 0x20 -> code\n
code < 0x400 -> if ((ch and 1) == 1) code shr 5 else code and 0x1f\n else ->\n when (ch % 3) {\n
2 -> code shr 10\n 1 -> (code shr 5) and 0x1f\n else -> code and 0x1f\n }\n }\n}\n\n/**\n
* Returns the Unicode general category of this character as an Int.\n */\ninternal fun Char.getCategoryValue(): Int
{\n val ch = this.code\n\n val index = binarySearchRange(Category.decodedRangeStart, ch)\n val start =

Category.decodedRangeStart[index]\n   val code = Category.decodedRangeCategory[index]\n   val value = categoryValueFrom(code, ch - start)\n\n   return if (value == 17) CharCategory.UNASSIGNED.value else value\n}\n\ninternal fun decodeVarLenBase64(base64: String, fromBase64: IntArray, resultLength: Int): IntArray {\n   val result = IntArray(resultLength)\n   var index = 0\n   var int = 0\n   var shift = 0\n   for (char in base64) {\n   val sixBit = fromBase64[char.code]\n   int = int or ((sixBit and 0x1f) shl shift)\n   if (sixBit < 0x20) {\n   result[index++] = int\n   int = 0\n   shift = 0\n   } else {\n   shift += 5\n   }\n   }\n   return result\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.js.*\nimport kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Reverses elements in the list in-place.\n */\npublic actual fun <T> MutableList<T>.reverse(): Unit {\n   val midPoint = (size / 2) - 1\n   if (midPoint < 0) return\n   var reverseIndex = lastIndex\n   for (index in 0..midPoint) {\n   val tmp = this[index]\n   this[index] = this[reverseIndex]\n   this[reverseIndex] = tmp\n   reverseIndex--\n   }\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateUnicodeData.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\n// 37 ranges totally\nprivate object Digit {\n   internal val rangeStart = intArrayOf(\n   0x0030, 0x0660, 0x06f0, 0x07c0, 0x0966, 0x09e6, 0x0a66, 0x0ae6, 0x0b66, 0x0be6, 0x0c66, 0x0ce6, 0x0d66, 0x0de6, 0x0e50, 0x0ed0, 0x0f20, 0x1040, 0x1090, 0x17e0, \n   0x1810, 0x1946, 0x19d0, 0x1a80, 0x1a90, 0x1b50, 0x1bb0, 0x1c40, 0x1c50, 0xa620, 0xa8d0, 0xa900, 0xa9d0, 0xa9f0, 0xaa50, 0xabf0, 0xff10, \n   )\n}\n\n/**\n * Returns the index of the largest element in [array] smaller or equal to the specified [needle],\n * or -1 if [needle] is smaller than the smallest element in [array].\n */\ninternal fun binarySearchRange(array: IntArray, needle: Int): Int {\n   var bottom = 0\n   var top = array.size - 1\n   var middle = -1\n   var value = 0\n   while (bottom <= top) {\n   middle = (bottom + top) / 2\n   value = array[middle]\n   if (needle > value)\n   bottom = middle + 1\n   else if (needle == value)\n   return middle\n   else\n   top = middle - 1\n   }\n   return middle - (if (needle < value) 1 else 0)\n}\n\n/**\n * Returns an integer from 0..9 indicating the digit this character represents,\n * or -1 if this character is not a digit.\n */\ninternal fun Char.digitToIntImpl(): Int {\n   val ch = this.code\n   val index = binarySearchRange(Digit.rangeStart, ch)\n   val diff = ch - Digit.rangeStart[index]\n   return if (diff < 10) diff else -1\n}\n\n/**\n * Returns `true` if this character is a digit.\n */\ninternal fun Char.isDigitImpl(): Boolean {\n   return digitToIntImpl() >= 0\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateUnicodeData.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\n// 222 ranges totally\nprivate object Letter {\n   val decodedRangeStart: IntArray\n   val decodedRangeLength: IntArray\n   val decodedRangeCategory: IntArray\n\n   init {\n   val toBase64 = \"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/\"\n   val fromBase64 = IntArray(128)\n   for (i in toBase64.indices) {\n   fromBase64[toBase64[i].code] = i\n   }\n\n   // rangeStartDiff.length = 356\n   val rangeStartDiff = \"hCgBpCQGYHZH5BRpBPPPPPPRMP5BPPlCPP6BkEPPPPcPXPzBvBrB3BOiDoBHwD+E3DauCnFmBmB2D6E1BlBTiBmBlBP5BhBiBrBvBjBqBnBPRtBiCmCtBlB0BmB5BiB7BmBgEmChBZgCoEoGVpBSfRhBPqKQ2BwBYoFgB4CJuTiEvBuCuDrF5DgEgFlJ1DgFmBQtBsBRGsB+BPiBlD1EIjDPRPPPQPPPPPGQSQS/DxENVNU+B9zCwBwBPPCkDPNnBPqDYY1R8B7FkFgTgwGgwUwmBgKwBuBScmEP/BPPPPPPrBP8B7F1B/ErBqC6B7BiBmBfQsBUwCw/KwqIwLwETPcPjQgJxFgBlBsD\"\n   val diff = decodeVarLenBase64(rangeStartDiff, fromBase64, 222)\n   val start = IntArray(diff.size)\n   for (i in diff.indices) {\n   if (i == 0) start[i] = diff[i]\n   else start[i] = start[i - 1] + diff[i]\n   }\n   decodedRangeStart = start\n\n   // rangeLength.length = 328\n   val rangeLength =

\"aaMBXHYH5BRpBPPPPPPRMP5BPPlCPPzBDOOPPcPXPzBvBjB3BOhDmBBpB7DoDYxB+EiBP1DoExBkBQhBekBPmBgBhBctBiBMWOOXhCsBpBkBUV3Ba4BkB0DlCgBXgBtD4FSdBfPhBPpKP0BvBXjEQ2CGsT8DhBtCqDpFvD1D3E0IrD2EkBJrBDOBsB+BPiBlB1EIjDPPPPPPPPPPPGPPMNLsBNPNPKCvBvBPPCkDPBmBPhDXXgD4B6FzEgDguG9vUtkB9JcuBSckEP/BPPPPPPBPf4FrBjEhBpC3B5BKaWPrBOwCk/KsCuLqDHPbPxPsFtEaaqDL\"\n decodedRangeLength = decodeVarLenBase64(rangeLength, fromBase64, 222)\n \n // rangeCategory.length = 959\n val rangeCategory = \"GFjgggUHGGFFZZZmzpz5qB6s6020B60ptltB6smt2sB60mz22B1+vv+8BZZ5s2850BW5q1ymtB506smzBF3q1q1qB1q1q1+Bgii4wDTm74g3KiggxqM60q1q1Bq1o1q1BF1qlrqrBZ2q5wprBGFZWWZGHFsjiooLowgmOowjkwCkgoiIk7ligGogiioBkwkiYkzj2oNoi+sbkwj04DghhkQ8wgiYkgoioDsgnkwC4gikQ//v+85BkwvoIsgoyI4yguI0whiwEowri4CoghsJowgqYowgm4DkwgsY/nwnzPowhmYkg6wI8yggZswikwHgxgmIoxgqYkwgk4DkxgmIkgoioBsgssoBgzgyI8g9gL8g9kI0wgwJoxgkoC0wgioFkw/wI0w53iF4gioYowjmgBHGq1qkgwBF1q1q8qBHwghuIwghyKk0goQkwgoQk3goQHGFHkyg0pBgxj6IoinkxDswno7Ikwhz9Bo0gioB8z48Rwli0xN0mpjoX8w78pDwltoqKHFGGwwgsIHFH3q1q16BFHWFZ1q10q1B2qlwq1B1q10q1B2q1yq1B6q1gq1Biq1qhxBir1qp1Bqt1q1qB1g1q1+B//3q16B///q1qBH/qlqq9Bholqq9B1i00a1q10qD1op1HkwmigEigiy6Cptogq1Bixo1kDq7/j00B2qgoBWGFm1lz50B6s5q1+BGWhggzhwBFFhgk4//Bo2jigE8wguI8wguI8wgugUog1qoB4qjmIwwi2KgkYHHH4lBgiFWkgIWoghssMmz5smrBZ3q1y50B5sm7gzBtz1smzB5smz50BqzqtmzB5sgzqzBF2/9//5BowgoIwmnkzPkwgk4C8ys65BkgoqI0wgy6FghquZo2giY0ghiIsgh24B4ghsQ8QF/v1q1OFs0O8iCHHF1qggz/B8wg6Iznv+//B08QgohsjK0QGFk7hsQ4gB\"\n decodedRangeCategory = decodeVarLenBase64(rangeCategory, fromBase64, 222)\n }\n}\n\n/**\n * Returns `true` if this character is a letter.\n */\ninternal fun Char.isLetterImpl(): Boolean {\n return getLetterType() != 0\n}\n\n/**\n * Returns `true` if this character is a lower case letter, or it has contributory property `Other_Lowercase`.\n */\ninternal fun Char.isLowerCaseImpl(): Boolean {\n return getLetterType() == 1 || code.isOtherLowercase()\n}\n\n/**\n * Returns `true` if this character is an upper case letter, or it has contributory property `Other_Uppercase`.\n */\ninternal fun Char.isUpperCaseImpl(): Boolean {\n return getLetterType() == 2 || code.isOtherUppercase()\n}\n\n/**\n * Returns\n * - `1` if the character is a lower case letter,\n * - `2` if the character is an upper case letter,\n * - `3` if the character is a letter but not a lower or upper case letter,\n * - `0` otherwise.\n */\nprivate fun Char.getLetterType(): Int {\n val ch = this.code\n val index = binarySearchRange(Letter.decodedRangeStart, ch)\n\n val rangeStart = Letter.decodedRangeStart[index]\n val rangeEnd = rangeStart + Letter.decodedRangeLength[index] - 1\n val code = Letter.decodedRangeCategory[index]\n\n if (ch > rangeEnd) {\n return 0\n }\n\n val lastTwoBits = code and 0x3\n\n if (lastTwoBits == 0) { // gap pattern\n var shift = 2\n var threshold = rangeStart\n for (i in 0..1) {\n threshold += (code shr shift) and 0x7f\n if (threshold > ch) {\n return 3\n }\n shift += 7\n threshold += (code shr shift) and 0x7f\n if (threshold > ch) {\n return 0\n }\n shift += 7\n }\n return 3\n }\n\n if (code <= 0x7) {\n return lastTwoBits\n }\n\n val distance = (ch - rangeStart)\n val shift = if (code <= 0x1F) distance % 2 else distance\n return (code shr (2 * shift)) and 0x3\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateUnicodeData.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nprivate object OtherLowercase {\n internal val otherLowerStart = intArrayOf(\n 0x00aa, 0x00ba, 0x02b0, 0x02c0, 0x02e0, 0x0345, 0x037a, 0x1d2c, 0x1d78, 0x1d9b, 0x2071, 0x207f, 0x2090, 0x2170, 0x24d0, 0x2c7c, 0xa69c, 0xa770, 0xa7f8, 0xab5c, \n)\n internal val otherLowerLength = intArrayOf(\n 1, 1, 9, 2, 5, 1, 1, 63, 1, 37, 1, 1, 13, 16, 26, 2, 2, 1, 2, 4, \n)\n}\n\ninternal fun Int.isOtherLowercase(): Boolean {\n val index = binarySearchRange(OtherLowercase.otherLowerStart, this)\n return index >= 0 && this < OtherLowercase.otherLowerStart[index] + OtherLowercase.otherLowerLength[index]\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateUnicodeData.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\ninternal fun Int.isOtherUppercase(): Boolean

{\n   return this in 0x2160..0x216f\n || this in 0x24b6..0x24cf\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.js.*\n\n/**\n * Returns a character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char sequence.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun CharSequence.elementAt(index: Int): Char {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, length: $length}\") }\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateUnicodeData.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\n// 4 ranges totally\ninternal fun Char.titlecaseCharImpl(): Char {\n   val code = this.code\n   // Letters repeating <Lu, Lt, Ll> sequence and code of the Lt is a multiple of 3, e.g. <\u01c4, \u01c5, \u01c6>\n   if (code in 0x01c4..0x01cc || code in 0x01f1..0x01f3) {\n   return (3 * ((code + 1) / 3)).toChar()\n   }\n   // Lower case letters whose title case mapping equivalent is equal to the original letter\n   if (code in 0x10d0..0x10fa || code in 0x10fd..0x10ff) {\n   return this\n   }\n   return uppercaseChar()\n}","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.js.*\nimport kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic actual fun UIntArray.elementAt(index: Int): UInt {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size}\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic actual fun ULongArray.elementAt(index: Int): ULong {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size}\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic actual fun UByteArray.elementAt(index: Int): UByte {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size}\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic actual fun UShortArray.elementAt(index: Int): UShort {\n   return elementAtOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, size: $size}\") }\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic actual fun UIntArray.asList(): List<UInt> {\n   return object : AbstractList<UInt>(), RandomAccess {\n   override val size: Int get() = this@asList.size\n   override fun isEmpty(): Boolean = this@asList.isEmpty()\n   override fun contains(element: UInt): Boolean = this@asList.contains(element)\n   override fun get(index: Int): UInt {\n   AbstractList.checkElementIndex(index, size)\n   return this@asList[index]\n   }\n   override fun indexOf(element: UInt): Int {\n   @Suppress(\"USELESS_CAST\")\n   if ((element as Any?) !is UInt) return -1\n   return this@asList.indexOf(element)\n   }\n   override fun lastIndexOf(element: UInt): Int {\n   @Suppress(\"USELESS_CAST\")\n   if ((element as Any?) !is UInt) return -1\n   return this@asList.lastIndexOf(element)\n   }\n   }\n}\n\n/**\n * Returns a [List] that wraps the original array.\n

```
*/\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic actual fun ULongArray.asList(): List<ULong> {\n    return object : AbstractList<ULong>(), RandomAccess {\n        override val size: Int get() = this@asList.size\n        override fun isEmpty(): Boolean = this@asList.isEmpty()\n        override fun contains(element: ULong): Boolean = this@asList.contains(element)\n        override fun get(index: Int): ULong {\n            AbstractList.checkElementIndex(index, size)\n            return this@asList[index]\n        }\n        override fun indexOf(element: ULong): Int {\n            @Suppress(\"USELESS_CAST\")\n            if ((element as Any?) !is ULong) return -1\n            return this@asList.indexOf(element)\n        }\n        override fun lastIndexOf(element: ULong): Int {\n            @Suppress(\"USELESS_CAST\")\n            if ((element as Any?) !is ULong) return -1\n            return this@asList.lastIndexOf(element)\n        }\n    }\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic actual fun UByteArray.asList(): List<UByte> {\n    return object : AbstractList<UByte>(), RandomAccess {\n        override val size: Int get() = this@asList.size\n        override fun isEmpty(): Boolean = this@asList.isEmpty()\n        override fun contains(element: UByte): Boolean = this@asList.contains(element)\n        override fun get(index: Int): UByte {\n            AbstractList.checkElementIndex(index, size)\n            return this@asList[index]\n        }\n        override fun indexOf(element: UByte): Int {\n            @Suppress(\"USELESS_CAST\")\n            if ((element as Any?) !is UByte) return -1\n            return this@asList.indexOf(element)\n        }\n        override fun lastIndexOf(element: UByte): Int {\n            @Suppress(\"USELESS_CAST\")\n            if ((element as Any?) !is UByte) return -1\n            return this@asList.lastIndexOf(element)\n        }\n    }\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic actual fun UShortArray.asList(): List<UShort> {\n    return object : AbstractList<UShort>(), RandomAccess {\n        override val size: Int get() = this@asList.size\n        override fun isEmpty(): Boolean = this@asList.isEmpty()\n        override fun contains(element: UShort): Boolean = this@asList.contains(element)\n        override fun get(index: Int): UShort {\n            AbstractList.checkElementIndex(index, size)\n            return this@asList[index]\n        }\n        override fun indexOf(element: UShort): Int {\n            @Suppress(\"USELESS_CAST\")\n            if ((element as Any?) !is UShort) return -1\n            return this@asList.indexOf(element)\n        }\n        override fun lastIndexOf(element: UShort): Int {\n            @Suppress(\"USELESS_CAST\")\n            if ((element as Any?) !is UShort) return -1\n            return this@asList.lastIndexOf(element)\n        }\n    }\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateUnicodeData.kt\n// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\n// 9 ranges totally\n/**\n * Returns `true` if this character is a whitespace.\n */\ninternal fun Char.isWhitespaceImpl(): Boolean {\n    val ch = this.code\n    return ch in 0x0009..0x000d\n            || ch in 0x001c..0x0020\n            || ch == 0x00a0\n            || ch > 0x1000 && (\n            ch == 0x1680\n            || ch in 0x2000..0x200a\n            || ch == 0x2028\n            || ch == 0x2029\n            || ch == 0x202f\n            || ch == 0x205f\n            || ch == 0x3000\n            )\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\n\npublic actual fun interface Comparator<T> {\n    @JsName(\"compare\")\n    public actual fun compare(a: T, b: T): Int\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js\n\nimport kotlin.annotation.AnnotationTarget.*\n\n@Target(FUNCTION)\n@Deprecated(\"Use inline extension function with body using dynamic\")\npublic annotation class nativeGetter\n\n@Target(FUNCTION)\n@Deprecated(\"Use inline extension function with body using dynamic\")\npublic annotation class nativeSetter\n\n@Target(FUNCTION)\n@Deprecated(\"Use inline extension function with body using dynamic\")\npublic annotation class nativeInvoke\n\n@Target(CLASS, FUNCTION, PROPERTY)\ninternal annotation class library(public val name: String = \"\")\n\n@Target(CLASS)\ninternal annotation class marker\n\n/**\n * Gives a declaration (a function, a property or a class) specific name in JavaScript.\n *\n * This may be useful in the following cases:\n *\n *   * There are two functions for which the
```

compiler gives same name in JavaScript, you can\n * mark one with `@JsName(...)` to prevent the compiler from reporting error.\n * * You are writing a JavaScript library in Kotlin. The compiler produces mangled names\n * for functions with parameters, which is unnatural for usual JavaScript developer.\n * You can put `@JsName(...)` on functions you want to be available from JavaScript.\n * * For some reason you want to rename declaration, e.g. there's common term in JavaScript\n * for a concept provided by the declaration, which in uncommon in Kotlin.\n *\n * Example:\n *\n * ``` kotlin\n * class Person(val name: String) {\n * fun hello() {\n * println(\"Hello $name!\")\n * }\n *\n * @JsName(\"helloWithGreeting\")\n * fun hello(greeting: String) {\n * println(\"$greeting $name!\")\n * }\n * }\n * ```\n *\n * @property name the name which compiler uses both for declaration itself and for all references to the declaration.\n * It's required to denote a valid JavaScript identifier.\n *\n */\n@Retention(AnnotationRetention.BINARY)\n@Target(CLASS, FUNCTION, PROPERTY, CONSTRUCTOR, PROPERTY_GETTER, PROPERTY_SETTER)\npublic actual annotation class JsName(actual val name: String)\n\n/**\n * Denotes an `external` declaration that must be imported from native JavaScript library.\n *\n * The compiler produces the code relevant for the target module system, for example, in case of CommonJS,\n * it will import the declaration via the `require(...)` function.\n *\n * The annotation can be used on top-level external declarations (classes, properties, functions) and files.\n * In case of file (which can't be `external`) the following rule applies: all the declarations in\n * the file must be `external`. By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object\n * that contain all the declarations from the file.\n *\n * Example:\n *\n * ``` kotlin\n * @JsModule(\"jquery\")\n * external abstract class JQuery() {\n * // some declarations here\n * }\n *\n * @JsModule(\"jquery\")\n * external fun JQuery(element: Element): JQuery\n * ```\n *\n * @property import name of a module to import declaration from.\n * It is not interpreted by the Kotlin compiler, it's passed as is directly to the target module system.\n *\n * @see JsNonModule\n */\n@Retention(AnnotationRetention.BINARY)\n@Target(CLASS, PROPERTY, FUNCTION, FILE)\npublic annotation class JsModule(val import: String)\n\n/**\n * Denotes an `external` declaration that can be used without module system.\n *\n * By default, an `external` declaration is available regardless your target module system.\n * However, by applying [JsModule] annotation you can make a declaration unavailable to *plain* module system.\n * Some JavaScript libraries are distributed both as a standalone downloadable piece of JavaScript and as a module available\n * as an npm package.\n * To tell the Kotlin compiler to accept both cases, you can augment [JsModule] with the `@JsNonModule` annotation.\n *\n * For example:\n *\n * ``` kotlin\n * @JsModule(\"jquery\")\n * @JsNonModule\n * @JsName(\"$\")\n * external abstract class JQuery() {\n * // some declarations here\n * }\n *\n * @JsModule(\"jquery\")\n * @JsNonModule\n * @JsName(\"$\")\n * external fun JQuery(element: Element): JQuery\n * ```\n *\n * @see JsModule\n */\n@Retention(AnnotationRetention.BINARY)\n@Target(CLASS, PROPERTY, FUNCTION, FILE)\npublic annotation class JsNonModule\n\n/**\n * Adds prefix to `external` declarations in a source file.\n *\n * JavaScript does not have concept of packages (namespaces). They are usually emulated by nested objects.\n * The compiler turns references to `external` declarations either to plain unprefixed names (in case of *plain* modules)\n * or to plain imports.\n * However, if a JavaScript library provides its declarations in packages, you won't be satisfied with this.\n * You can tell the compiler to generate additional prefix before references to `external` declarations using the `@JsQualifier(...)`\n * annotation.\n *\n * Note that a file marked with the `@JsQualifier(...)` annotation can't contain non-`external` declarations.\n *\n * Example:\n *\n * ```\n * @file:JsQualifier(\"my.jsPackageName\")\n * package some.kotlinPackage\n *\n * external fun foo(x: Int)\n *\n * external fun bar(): String\n * ```\n *\n * @property value the qualifier to add to the declarations in the generated code.\n * It must be a sequence of valid JavaScript identifiers separated by the `.` character.\n * Examples of valid qualifiers are: `foo`, `bar.Baz`, `_.$0.f`.\n *\n * @see JsModule\n */\n@Retention(AnnotationRetention.BINARY)\n@Target(AnnotationTarget.FILE)\npublic annotation class JsQualifier(val value: String)\n\n/**\n * Exports top-level declaration on JS platform.\n *\n * Compiled module exposes declarations that are marked with this annotation without name mangling.\n *\n * This annotation can be applied to either files or top-level declarations.\n *\n * It is currently prohibited to export the following kinds of declarations:\n *\n * * `expect` declarations\n * * inline functions with reified type parameters\n * * suspend functions\n * * secondary constructors without `@JsName`\n * * extension properties\n * * enum classes\n * *

annotation classes\n *\n * Signatures of exported declarations must only contain \"exportable\" types:\n *\n * * `dynamic`, `Any`, `String`, `Boolean`, `Byte`, `Short`, `Int`, `Float`, `Double`\n * * `BooleanArray`, `ByteArray`, `ShortArray`, `IntArray`, `FloatArray`, `DoubleArray`\n * * `Array<exportable-type>`\n * * Function types with exportable parameters and return types\n * * `external` or `@JsExport` classes and interfaces\n * * Nullable counterparts of types above\n * * Unit return type. Must not be nullable\n *\n * This annotation is experimental, meaning that restrictions mentioned above are subject to change.\n */\n@ExperimentalJsExport\n@Retention(AnnotationRetention.BINARY)\n@Target(CLASS, PROPERTY, FUNCTION, FILE)\n@SinceKotlin(\"1.3\")\npublic actual annotation class JsExport\n\n/**\n * Forces a top-level property to be initialized eagerly, opposed to lazily on the first access to file and/or property.\n */\n@ExperimentalStdlibApi\n@Retention(AnnotationRetention.BINARY)\n@Target(AnnotationTarget.PROPERTY)\n@SinceKotlin(\"1.6\")\n@Deprecated(\"This annotation is a temporal migration assistance and may be removed in the future releases, please consider filing an issue about the case where it is needed\")\npublic annotation class EagerInitialization\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.jvm\n\n// these are used in common generated code in stdlib\n\n// TODO: find how to deprecate these ones\n\n@Target(AnnotationTarget.FIELD)\n@Retention(AnnotationRetention.SOURCE)\npublic actual annotation class Volatile\n\n@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY_GETTER, AnnotationTarget.PROPERTY_SETTER)\n@Retention(AnnotationRetention.SOURCE)\npublic actual annotation class Synchronized\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n/**\n * Provides a skeletal implementation of the [MutableCollection] interface.\n *\n * @param E the type of elements contained in the collection. The collection is invariant in its element type.\n */\npublic actual abstract class AbstractMutableCollection<E> protected actual constructor() : AbstractCollection<E>(), MutableCollection<E> {\n\n   actual abstract override fun add(element: E): Boolean\n\n   actual override fun remove(element: E): Boolean {\n   checkIsMutable()\n   val iterator = iterator()\n   while (iterator.hasNext()) {\n   if (iterator.next() == element) {\n   iterator.remove()\n   return true\n   }\n   }\n   return false\n   }\n\n   actual override fun addAll(elements: Collection<E>): Boolean {\n   checkIsMutable()\n   var modified = false\n   for (element in elements) {\n   if (add(element)) modified = true\n   }\n   return modified\n   }\n\n   actual override fun removeAll(elements: Collection<E>): Boolean {\n   checkIsMutable()\n   return (this as MutableIterable<E>).removeAll { it in elements }\n   }\n\n   actual override fun retainAll(elements: Collection<E>): Boolean {\n   checkIsMutable()\n   return (this as MutableIterable<E>).removeAll { it !in elements }\n   }\n\n   actual override fun clear(): Unit {\n   checkIsMutable()\n   val iterator = this.iterator()\n   while (iterator.hasNext()) {\n   iterator.next()\n   iterator.remove()\n   }\n   }\n\n   @Deprecated(\"Provided so that subclasses inherit this function\", level = DeprecationLevel.HIDDEN)\n   @JsName(\"toJSON\")\n   protected fun toJSON(): Any = this.toArray()\n\n\n   /**\n   * This method is called every time when a mutating method is called on this mutable collection.\n   * Mutable collections that are built (frozen) must throw `UnsupportedOperationException`.\n   */\n   internal open fun checkIsMutable(): Unit { }\n}\n\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n/*\n * Based on GWT AbstractList\n * Copyright 2007 Google Inc.\n*/\n\n\npackage kotlin.collections\n\n/**\n * Provides a skeletal implementation of the [MutableList] interface.\n *\n * @param E the type of elements contained in the list. The list is invariant in its element type.\n */\npublic actual abstract class AbstractMutableList<E> protected actual constructor() : AbstractMutableCollection<E>(), MutableList<E> {\n   protected var modCount: Int = 0\n\n   abstract override fun add(index: Int, element: E): Unit\n   abstract override fun removeAt(index: Int): E\n   abstract override fun set(index: Int, element: E): E\n\n   /**\n   * Adds the specified element to the end of this list.\n   *\n   * @return `true` because the list is always modified as the result

```
of this operation.\n     */\n    actual override fun add(element: E): Boolean {\n        checkIsMutable()\n        add(size, element)\n        return true\n    }\n\n    actual override fun addAll(index: Int, elements: Collection<E>): Boolean {\n        AbstractList.checkPositionIndex(index, size)\n\n        checkIsMutable()\n        var _index = index\n        var changed = false\n        for (e in elements) {\n            add(_index++, e)\n            changed = true\n        }\n        return changed\n    }\n\n    actual override fun clear() {\n        checkIsMutable()\n        removeRange(0, size)\n    }\n\n    actual override fun removeAll(elements: Collection<E>): Boolean {\n        checkIsMutable()\n        return removeAll { it in elements }\n    }\n\n    actual override fun retainAll(elements: Collection<E>): Boolean {\n        checkIsMutable()\n        return removeAll { it !in elements }\n    }\n\n\n    actual override fun iterator(): MutableIterator<E> = IteratorImpl()\n\n    actual override fun contains(element: E): Boolean = indexOf(element) >= 0\n\n    actual override fun indexOf(element: E): Int {\n        for (index in 0..lastIndex) {\n            if (get(index) == element) {\n                return index\n            }\n        }\n        return -1\n    }\n\n    actual override fun lastIndexOf(element: E): Int {\n        for (index in lastIndex downTo 0) {\n            if (get(index) == element) {\n                return index\n            }\n        }\n        return -1\n    }\n\n    actual override fun listIterator(): MutableListIterator<E> = listIterator(0)\n    actual override fun listIterator(index: Int): MutableListIterator<E> = ListIteratorImpl(index)\n\n\n    actual override fun subList(fromIndex: Int, toIndex: Int): MutableList<E> = SubList(this, fromIndex, toIndex)\n\n    /**\n     * Removes the range of elements from this list starting from [fromIndex] and ending with but not including [toIndex].\n     */\n    protected open fun removeRange(fromIndex: Int, toIndex: Int) {\n        val iterator = listIterator(fromIndex)\n        repeat(toIndex - fromIndex) {\n            iterator.next()\n            iterator.remove()\n        }\n    }\n\n    /**\n     * Compares this list with another list instance with the ordered structural equality.\n     *\n     * @return true, if [other] instance is a [List] of the same size, which contains the same elements in the same order.\n     */\n    override fun equals(other: Any?): Boolean {\n        if (other === this) return true\n        if (other !is List<*>) return false\n\n        return AbstractList.orderedEquals(this, other)\n    }\n\n    /**\n     * Returns the hash code value for this list.\n     */\n    override fun hashCode(): Int = AbstractList.orderedHashCode(this)\n\n\n    private open inner class IteratorImpl : MutableIterator<E> {\n        /** the index of the item that will be returned on the next call to [next]`()` */\n        protected var index = 0\n        /** the index of the item that was returned on the previous call to [next]`()`\n         * or [ListIterator.previous]`()` (for `ListIterator`),\n         * -1 if no such item exists\n         */\n        protected var last = -1\n\n        override fun hasNext(): Boolean = index < size\n\n        override fun next(): E {\n            if (!hasNext()) throw NoSuchElementException()\n            last = index++\n            return get(last)\n        }\n\n        override fun remove() {\n            check(last != -1) { \"Call next() or previous() before removing element from the iterator.\" }\n\n            removeAt(last)\n            index = last\n            last = -1\n        }\n    }\n\n    /**\n     * Implementation of `MutableListIterator` for abstract lists.\n     */\n    private inner class ListIteratorImpl(index: Int) : IteratorImpl(), MutableListIterator<E> {\n\n        init {\n            AbstractList.checkPositionIndex(index, this@AbstractMutableList.size)\n            this.index = index\n        }\n\n        override fun hasPrevious(): Boolean = index > 0\n\n        override fun nextIndex(): Int = index\n\n        override fun previous(): E {\n            if (!hasPrevious()) throw NoSuchElementException()\n\n            last = --index\n            return get(last)\n        }\n\n        override fun previousIndex(): Int = index - 1\n\n        override fun add(element: E) {\n            add(index, element)\n            index++\n            last = -1\n        }\n\n        override fun set(element: E) {\n            check(last != -1) { \"Call next() or previous() before updating element value with the iterator.\" }\n            set(last, element)\n        }\n    }\n\n    private class SubList<E>(private val list: AbstractMutableList<E>, private val fromIndex: Int, toIndex: Int) : AbstractMutableList<E>(), RandomAccess {\n        private var _size: Int = 0\n\n        init {\n            AbstractList.checkRangeIndexes(fromIndex, toIndex, list.size)\n            this._size = toIndex - fromIndex\n        }\n\n        override fun add(index: Int, element: E) {\n            AbstractList.checkPositionIndex(index, _size)\n\n            list.add(fromIndex + index, element)\n            _size++\n        }\n\n        override fun get(index: Int): E {\n            AbstractList.checkElementIndex(index, _size)\n\n            return list[fromIndex + index]\n        }\n\n        override fun removeAt(index: Int): E {\n            AbstractList.checkElementIndex(index, _size)\n\n            val result = list.removeAt(fromIndex + index)\n            _size--\n            return result\n        }\n\n        override fun set(index: Int, element: E): E {\n            AbstractList.checkElementIndex(index, _size)\n\n            return list.set(fromIndex + index,
```

element)\n }\n\n override val size: Int get() = _size\n\n internal override fun checkIsMutable(): Unit = list.checkIsMutable()\n }\n\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n/*\n * Based on GWT AbstractMap\n * Copyright 2007 Google Inc.\n */\n\npackage kotlin.collections\n\n/**\n * Provides a skeletal implementation of the [MutableMap] interface.\n *\n * The implementor is required to implement [entries] property, which should return mutable set of map entries, and [put] function.\n *\n * @param K the type of map keys. The map is invariant in its key type.\n * @param V the type of map values. The map is invariant in its value type.\n */\npublic actual abstract class AbstractMutableMap<K, V> protected actual constructor() : AbstractMap<K, V>(), MutableMap<K, V> {\n\n /**\n * A mutable [Map.Entry] shared by several [Map] implementations.\n */\n internal open class SimpleEntry<K, V>(override val key: K, value: V) : MutableMap.MutableEntry<K, V> {\n constructor(entry: Map.Entry<K, V>) : this(entry.key, entry.value)\n\n private var _value = value\n\n override val value: V get() = _value\n\n override fun setValue(newValue: V): V {\n // Should check if the map containing this entry is mutable.\n // However, to not increase entry memory footprint it might be worthwhile not to check it here and\n // force subclasses that implement `build()` (freezing) operation to implement their own `MutableEntry`.\n// this@AbstractMutableMap.checkIsMutable()\n val oldValue = this._value\n this._value = newValue\n return oldValue\n }\n\n override fun hashCode(): Int = entryHashCode(this)\n override fun toString(): String = entryToString(this)\n override fun equals(other: Any?): Boolean = entryEquals(this, other)\n\n }\n\n // intermediate abstract class to workaround KT-43321\n internal abstract class AbstractEntrySet<E : Map.Entry<K, V>, K, V> : AbstractMutableSet<E>() {\n final override fun contains(element: E): Boolean = containsEntry(element)\n abstract fun containsEntry(element: Map.Entry<K, V>): Boolean\n final override fun remove(element: E): Boolean = removeEntry(element)\n abstract fun removeEntry(element: Map.Entry<K, V>): Boolean\n }\n\n actual override fun clear() {\n entries.clear()\n }\n\n private var _keys: MutableSet<K>? = null\n actual override val keys: MutableSet<K>\n get() {\n if (_keys == null) {\n _keys = object : AbstractMutableSet<K>() {\n override fun add(element: K): Boolean = throw UnsupportedOperationException(\"Add is not supported on keys\")\n override fun clear() {\n this@AbstractMutableMap.clear()\n }\n\n override operator fun contains(element: K): Boolean = containsKey(element)\n\n override operator fun iterator(): MutableIterator<K> {\n val entryIterator = entries.iterator()\n return object : MutableIterator<K> {\n override fun hasNext(): Boolean = entryIterator.hasNext()\n override fun next(): K = entryIterator.next().key\n override fun remove() = entryIterator.remove()\n }\n }\n\n override fun remove(element: K): Boolean {\n checkIsMutable()\n if (containsKey(element)) {\n this@AbstractMutableMap.remove(element)\n return true\n }\n return false\n }\n\n override val size: Int get() = this@AbstractMutableMap.size\n\n override fun checkIsMutable(): Unit = this@AbstractMutableMap.checkIsMutable()\n }\n }\n return _keys!!\n }\n\n actual abstract override fun put(key: K, value: V): V?\n\n actual override fun putAll(from: Map<out K, V>) {\n checkIsMutable()\n for ((key, value) in from) {\n put(key, value)\n }\n }\n\n private var _values: MutableCollection<V>? = null\n actual override val values: MutableCollection<V>\n get() {\n if (_values == null) {\n _values = object : AbstractMutableCollection<V>() {\n override fun add(element: V): Boolean = throw UnsupportedOperationException(\"Add is not supported on values\")\n override fun clear() = this@AbstractMutableMap.clear()\n\n override operator fun contains(element: V): Boolean = containsValue(element)\n\n override operator fun iterator(): MutableIterator<V> {\n val entryIterator = entries.iterator()\n return object : MutableIterator<V> {\n override fun hasNext(): Boolean = entryIterator.hasNext()\n override fun next(): V = entryIterator.next().value\n override fun remove() = entryIterator.remove()\n }\n }\n\n override val size: Int get() = this@AbstractMutableMap.size\n\n override fun checkIsMutable(): Unit =

this@AbstractMutableMap.checkIsMutable()\n }\n }\n return _values!!\n }\n\n actual override fun remove(key: K): V? {\n checkIsMutable()\n val iter = entries.iterator()\n while (iter.hasNext()) {\n val entry = iter.next()\n val k = entry.key\n if (key == k) {\n val value = entry.value\n iter.remove()\n return value\n }\n }\n return null\n }\n\n\n /**\n * This method is called every time when a mutating method is called on this mutable map.\n * Mutable maps that are built (frozen) must throw `UnsupportedOperationException`.\n */\n internal open fun checkIsMutable(): Unit {}\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\npackage kotlin.collections\n\n/**\n * Provides a skeletal implementation of the [MutableSet] interface.\n *\n * @param E the type of elements contained in the set. The set is invariant in its element type.\n */\npublic actual abstract class AbstractMutableSet<E> protected actual constructor() : AbstractMutableCollection<E>(), MutableSet<E> {\n\n /**\n * Compares this set with another set instance with the unordered structural equality.\n *\n * @return `true`, if [other] instance is a [Set] of the same size, all elements of which are contained in this set.\n */\n override fun equals(other: Any?): Boolean {\n if (other === this) return true\n if (other !is Set<*>) return false\n return AbstractSet.setEquals(this, other)\n }\n\n /**\n * Returns the hash code value for this set.\n */\n override fun hashCode(): Int = AbstractSet.unorderedHashCode(this)\n\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n/**\n * Provides a [MutableList] implementation, which uses a resizable array as its backing storage.\n *\n * This implementation doesn't provide a way to manage capacity, as backing JS array is resizeable itself.\n * There is no speed advantage to pre-allocating array sizes in JavaScript, so this implementation does not include any of the\n * capacity and \"growth increment\" concepts.\n */\npublic actual open class ArrayList<E> internal constructor(private var array: Array<Any?>) : AbstractMutableList<E>(), MutableList<E>, RandomAccess {\n private var isReadOnly: Boolean = false\n\n /**\n * Creates an empty [ArrayList].\n */\n public actual constructor() : this(emptyArray()) {}\n\n /**\n * Creates an empty [ArrayList].\n * @param initialCapacity initial capacity (ignored)\n */\n public actual constructor(initialCapacity: Int) : this(emptyArray()) {}\n\n /**\n * Creates an [ArrayList] filled from the [elements] collection.\n */\n public actual constructor(elements: Collection<E>) : this(elements.toTypedArray<Any?>()) {}\n\n @PublishedApi\n internal fun build(): List<E> {\n checkIsMutable()\n isReadOnly = true\n return this\n }\n\n /** Does nothing in this ArrayList implementation. */\n public actual fun trimToSize() {}\n\n /** Does nothing in this ArrayList implementation. */\n public actual fun ensureCapacity(minCapacity: Int) {}\n\n actual override val size: Int get() = array.size\n\n @Suppress(\"UNCHECKED_CAST\")\n actual override fun get(index: Int): E = array[rangeCheck(index)] as E\n\n actual override fun set(index: Int, element: E): E {\n checkIsMutable()\n rangeCheck(index)\n\n @Suppress(\"UNCHECKED_CAST\")\n return array[index].apply { array[index] = element } as E\n }\n\n actual override fun add(element: E): Boolean {\n checkIsMutable()\n array.asDynamic().push(element)\n modCount++\n return true\n }\n\n actual override fun add(index: Int, element: E): Unit {\n checkIsMutable()\n array.asDynamic().splice(insertionRangeCheck(index), 0, element)\n modCount++\n }\n\n actual override fun addAll(elements: Collection<E>): Boolean {\n checkIsMutable()\n if (elements.isEmpty()) return false\n\n array += elements.toTypedArray<Any?>()\n modCount++\n return true\n }\n\n actual override fun addAll(index: Int, elements: Collection<E>): Boolean {\n checkIsMutable()\n insertionRangeCheck(index)\n\n if (index == size) return addAll(elements)\n if (elements.isEmpty()) return false\n when (index) {\n size -> return addAll(elements)\n 0 -> array = elements.toTypedArray<Any?>() + array\n else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))\n }\n\n modCount++\n return true\n }\n\n actual override fun removeAt(index: Int): E {\n checkIsMutable()\n rangeCheck(index)\n modCount++\n return if (index == lastIndex)\n array.asDynamic().pop()\n else\n array.asDynamic().splice(index, 1)[0]\n }\n\n actual override fun remove(element: E): Boolean {\n

```
checkIsMutable()\n        for (index in array.indices) {\n            if (array[index] == element) {\n
array.asDynamic().splice(index, 1)\n                modCount++\n                return true\n            }\n        }\n        return
false\n    }\n\n    override fun removeRange(fromIndex: Int, toIndex: Int) {\n        checkIsMutable()\n
modCount++\n        array.asDynamic().splice(fromIndex, toIndex - fromIndex)\n    }\n\n    actual override fun
clear() {\n        checkIsMutable()\n        array = emptyArray()\n        modCount++\n    }\n\n\n    actual override fun
indexOf(element: E): Int = array.indexOf(element)\n\n    actual override fun lastIndexOf(element: E): Int =
array.lastIndexOf(element)\n\n    override fun toString() = arrayToString(array)\n\n
@Suppress(\"UNCHECKED_CAST\")\n    override fun <T> toArray(array: Array<T>): Array<T> {\n        if
(array.size < size) {\n            return toArray() as Array<T>\n        }\n\n        (this.array as
Array<T>).copyInto(array)\n\n        if (array.size > size) {\n            array[size] = null as T // null-terminate\n
}\n\n        return array\n    }\n\n    override fun toArray(): Array<Any?> {\n        return js(\"[]\").slice.call(array)\n
}\n\n\n    internal override fun checkIsMutable() {\n        if (isReadOnly) throw UnsupportedOperationException()\n
  }\n\n    private fun rangeCheck(index: Int) = index.apply {\n        AbstractList.checkElementIndex(index, size)\n
}\n\n    private fun insertionRangeCheck(index: Int) = index.apply {\n        AbstractList.checkPositionIndex(index,
size)\n    }\n}","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n *
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*/\n\npackage kotlin.collections\n\ninternal fun <T> sortArrayWith(array: Array<out T>, comparison: (T, T) -> Int)
{\n    if (getStableSortingIsSupported()) {\n        array.asDynamic().sort(comparison)\n    } else {\n
mergeSort(array.unsafeCast<Array<T>>(), 0, array.lastIndex, Comparator(comparison))\n    }\n}\n\ninternal fun
<T> sortArrayWith(array: Array<out T>, comparator: Comparator<in T>) {\n    if (getStableSortingIsSupported())
{\n        val comparison = { a: T, b: T -> comparator.compare(a, b) }\n        array.asDynamic().sort(comparison)\n
} else {\n        mergeSort(array.unsafeCast<Array<T>>(), 0, array.lastIndex, comparator)\n    }\n}\n\ninternal fun
<T> sortArrayWith(array: Array<out T>, fromIndex: Int, toIndex: Int, comparator: Comparator<in T>) {\n    if
(fromIndex < toIndex - 1) {\n        mergeSort(array.unsafeCast<Array<T>>(), fromIndex, toIndex - 1, comparator)\n
  }\n}\n\ninternal fun <T : Comparable<T>> sortArray(array: Array<out T>) {\n    if
(getStableSortingIsSupported()) {\n        val comparison = { a: T, b: T -> a.compareTo(b) }\n
array.asDynamic().sort(comparison)\n    } else {\n        mergeSort(array.unsafeCast<Array<T>>(), 0,
array.lastIndex, naturalOrder())\n    }\n}\n\nprivate var _stableSortingIsSupported: Boolean? = null\nprivate fun
getStableSortingIsSupported(): Boolean {\n    _stableSortingIsSupported?.let { return it }\n
_stableSortingIsSupported = false\n\n    val array = js(\"[]\").unsafeCast<Array<Int>>()\n    // known
implementations may use stable sort for arrays of up to 512 elements\n    // so we create slightly more elements to
test stability\n    for (index in 0 until 600) array.asDynamic().push(index)\n    val comparison = { a: Int, b: Int -> (a
and 3) - (b and 3) }\n    array.asDynamic().sort(comparison)\n    for (index in 1 until array.size) {\n        val a =
array[index - 1]\n        val b = array[index]\n        if ((a and 3) == (b and 3) && a >= b) return false\n    }\n
_stableSortingIsSupported = true\n    return true\n}\n\n\nprivate fun <T> mergeSort(array: Array<T>, start: Int,
endInclusive: Int, comparator: Comparator<in T>) {\n    val buffer =
arrayOfNulls<Any?>(array.size).unsafeCast<Array<T>>()\n    val result = mergeSort(array, buffer, start,
endInclusive, comparator)\n    if (result !== array) {\n        for (i in start..endInclusive) array[i] = result[i]\n
}\n}\n\n// Both start and end are inclusive indices.\nprivate fun <T> mergeSort(array: Array<T>, buffer: Array<T>,
start: Int, end: Int, comparator: Comparator<in T>): Array<T> {\n    if (start == end) {\n        return array\n    }\n\n
val median = (start + end) / 2\n    val left = mergeSort(array, buffer, start, median, comparator)\n    val right =
mergeSort(array, buffer, median + 1, end, comparator)\n\n    val target = if (left === buffer) array else buffer\n\n    //
Merge.\n    var leftIndex = start\n    var rightIndex = median + 1\n    for (i in start..end) {\n        when {\n
leftIndex <= median && rightIndex <= end -> {\n                val leftValue = left[leftIndex]\n                val rightValue
= right[rightIndex]\n\n                if (comparator.compare(leftValue, rightValue) <= 0) {\n                    target[i] =
leftValue\n                    leftIndex++\n                } else {\n                    target[i] = rightValue\n
rightIndex++\n                }\n            }\n            leftIndex <= median -> {\n                target[i] = left[leftIndex]\n
    leftIndex++\n            }\n            else /* rightIndex <= end */ -> {\n                target[i] = right[rightIndex]\n
```

rightIndex++\n Unit // TODO: Fix KT-31506\n }\n }\n }\n\n return target\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n\n@OptIn(ExperimentalUnsignedTypes::class)\n@SinceKotlin(\"1.3\")\n@kotlin.js.JsName(\"contentDeepHashCodeImpl\")\ninternal fun <T> Array<out T>?.contentDeepHashCodeImpl(): Int {\n if (this == null) return 0\n var result = 1\n for (element in this) {\n val elementHash = when {\n element == null -> 0\n isArrayish(element) -> (element.unsafeCast<Array<*>>()).contentDeepHashCodeImpl()\n\n element is UByteArray -> element.contentHashCode()\n element is UShortArray -> element.contentHashCode()\n element is UIntArray -> element.contentHashCode()\n element is ULongArray -> element.contentHashCode()\n\n else -> element.hashCode()\n }\n\n result = 31 * result + elementHash\n }\n return result\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\ninternal interface EqualityComparator {\n /**\n * Subclasses must override to return a value indicating\n * whether or not two keys or values are equal.\n */\n abstract fun equals(value1: Any?, value2: Any?): Boolean\n\n /**\n * Subclasses must override to return the hash code of a given key.\n */\n abstract fun getHashCode(value: Any?): Int\n\n\n object HashCode : EqualityComparator {\n override fun equals(value1: Any?, value2: Any?): Boolean = value1 == value2\n\n override fun getHashCode(value: Any?): Int = value?.hashCode() ?: 0\n }\n}","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n/*\n * Based on GWT AbstractHashMap\n * Copyright 2008 Google Inc.\n */\n\npackage kotlin.collections\n\nimport kotlin.collections.MutableMap.MutableEntry\n\n/**\n * Hash table based implementation of the [MutableMap] interface.\n *\n * This implementation makes no guarantees regarding the order of enumeration of [keys], [values] and [entries] collections.\n */\n// Classes that extend HashMap and implement `build()` (freezing) operation\n// have to make sure mutating methods check `checkIsMutable`.\npublic actual open class HashMap<K, V> : AbstractMutableMap<K, V>, MutableMap<K, V> {\n\n private inner class EntrySet : AbstractEntrySet<MutableEntry<K, V>, K, V>() {\n\n override fun add(element: MutableEntry<K, V>): Boolean = throw UnsupportedOperationException(\"Add is not supported on entries\")\n override fun clear() {\n this@HashMap.clear()\n }\n\n override fun containsEntry(element: Map.Entry<K, V>): Boolean = this@HashMap.containsEntry(element)\n\n override operator fun iterator(): MutableIterator<MutableEntry<K, V>> = internalMap.iterator()\n\n override fun removeEntry(element: Map.Entry<K, V>): Boolean {\n if (contains(element)) {\n this@HashMap.remove(element.key)\n return true\n }\n return false\n }\n\n override val size: Int get() = this@HashMap.size\n }\n\n\n /**\n * Internal implementation of the map: either string-based or hashcode-based.\n */\n private val internalMap: InternalMap<K, V>\n\n private val equality: EqualityComparator\n\n internal constructor(internalMap: InternalMap<K, V>) : super() {\n this.internalMap = internalMap\n this.equality = internalMap.equality\n }\n\n /**\n * Constructs an empty [HashMap] instance.\n */\n actual constructor() : this(InternalHashCodeMap(EqualityComparator.HashCode))\n\n /**\n * Constructs an empty [HashMap] instance.\n *\n * @param initialCapacity the initial capacity (ignored)\n * @param loadFactor the load factor (ignored)\n *\n * @throws IllegalArgumentException if the initial capacity or load factor are negative\n */\n actual constructor(initialCapacity: Int, loadFactor: Float) : this() {\n // This implementation of HashMap has no need of load factors or capacities.\n require(initialCapacity >= 0) { \"Negative initial capacity: $initialCapacity\" }\n require(loadFactor >= 0) { \"Non-positive load factor: $loadFactor\" }\n }\n\n actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)\n\n\n /**\n * Constructs an instance of [HashMap] filled with the contents of the specified [original] map.\n */\n actual constructor(original: Map<out K, V>) : this() {\n this.putAll(original)\n }\n\n actual override fun clear() {\n internalMap.clear()\n// structureChanged(this)\n }\n\n actual override fun containsKey(key: K): Boolean = internalMap.contains(key)\n\n actual override fun containsValue(value: V): Boolean = internalMap.any {

equality.equals(it.value, value) }\n\n   private var _entries: MutableSet<MutableMap.MutableEntry<K, V>>? = null\n   actual override val entries: MutableSet<MutableMap.MutableEntry<K, V>>\n   get() {\n   if (_entries == null) {\n   _entries = createEntrySet()\n   }\n   return _entries!!\n   }\n\n   internal open fun createEntrySet(): MutableSet<MutableMap.MutableEntry<K, V>> = EntrySet()\n\n   actual override operator fun get(key: K): V? = internalMap.get(key)\n\n   actual override fun put(key: K, value: V): V? = internalMap.put(key, value)\n\n   actual override fun remove(key: K): V? = internalMap.remove(key)\n\n   actual override val size: Int get() = internalMap.size\n\n}\n\n/**\n * Constructs the specialized implementation of [HashMap] with [String] keys, which stores the keys as properties of\n * JS object without hashing them.\n */\npublic fun <V> stringMapOf(vararg pairs: Pair<String, V>): HashMap<String, V> {\n   return HashMap<String, V>(InternalStringMap(EqualityComparator.HashCode)).apply { putAll(pairs) }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n/*\n * Based on GWT HashSet\n * Copyright 2008 Google Inc.\n */\n\npackage kotlin.collections\n\n/**\n * The implementation of the [MutableSet] interface, backed by a [HashMap] instance.\n */\n// Classes that extend HashSet and implement `build()` (freezing) operation\n// have to make sure mutating methods check `checkIsMutable`.\npublic actual open class HashSet<E> : AbstractMutableSet<E>, MutableSet<E> {\n\n   internal val map: HashMap<E, Any>\n\n   /**\n   * Constructs a new empty [HashSet].\n   */\n   actual constructor() {\n   map = HashMap<E, Any>()\n   }\n\n   /**\n   * Constructs a new [HashSet] filled with the elements of the specified collection.\n   */\n   actual constructor(elements: Collection<E>) {\n   map = HashMap<E, Any>(elements.size)\n   addAll(elements)\n   }\n\n   /**\n   * Constructs a new empty [HashSet].\n   *\n   * @param initialCapacity the initial capacity (ignored)\n   * @param loadFactor   the load factor (ignored)\n   *\n   * @throws IllegalArgumentException if the initial capacity or load factor are negative\n   */\n   actual constructor(initialCapacity: Int, loadFactor: Float) {\n   map = HashMap<E, Any>(initialCapacity, loadFactor)\n   }\n\n   actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)\n\n   /**\n   * Protected constructor to specify the underlying map. This is used by\n   * LinkedHashSet.\n\n   * @param map underlying map to use.\n   */\n   internal constructor(map: HashMap<E, Any>) {\n   this.map = map\n   }\n\n   actual override fun add(element: E): Boolean {\n   val old = map.put(element, this)\n   return old == null\n   }\n\n   actual override fun clear() {\n   map.clear()\n   }\n\n// public override fun clone(): Any {\n//   return HashSet<E>(this)\n// }\n\n   actual override operator fun contains(element: E): Boolean = map.containsKey(element)\n\n   actual override fun isEmpty(): Boolean = map.isEmpty()\n\n   actual override fun iterator(): MutableIterator<E> = map.keys.iterator()\n\n   actual override fun remove(element: E): Boolean = map.remove(element) != null\n\n   actual override val size: Int get() = map.size\n\n}\n\n/**\n * Creates a new instance of the specialized implementation of [HashSet] with the specified [String] elements,\n * which elements the keys as properties of JS object without hashing them.\n */\npublic fun stringSetOf(vararg elements: String): HashSet<String> {\n   return HashSet(stringMapOf<Any>()).apply { addAll(elements) }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n/*\n * Based on GWT InternalHashCodeMap\n * Copyright 2008 Google Inc.\n */\n\npackage kotlin.collections\n\nimport kotlin.collections.MutableMap.MutableEntry\nimport kotlin.collections.AbstractMutableMap.SimpleEntry\n\n/**\n * A simple wrapper around JavaScriptObject to provide [java.util.Map]-like semantics for any\n * key type.\n *\n *\n * Implementation notes:\n *\n *\n * A key's hashCode is the index in backingMap which should contain that key. Since several keys may\n * have the same hash, each value in hashCodeMap is actually an array containing all entries whose\n * keys share the same hash.\n */\ninternal class InternalHashCodeMap<K, V>(override val equality: EqualityComparator) : InternalMap<K, V> {\n\n   private var backingMap: dynamic = createJsMap()\n   override var size: Int = 0\n   private set\n\n   override fun put(key: K, value: V): V? {\n   val hashCode = equality.getHashCode(key)\n   val chainOrEntry = getChainOrEntryOrNull(hashCode)\n   if (chainOrEntry == null) {\n   // This is a new chain, put it to the map.\n   backingMap[hashCode] = SimpleEntry(key, value)\n   } else {\n   if (chainOrEntry !is Array<*>) {\n   // It is an entry\n   val entry: SimpleEntry<K, V> = chainOrEntry\n   if

(equality.equals(entry.key, key)) {\n return entry.setValue(value)\n } else {\n backingMap[hashCode] = arrayOf(entry, SimpleEntry(key, value))\n size++\n return null\n }\n } else {\n // Chain already exists, perhaps key also exists.\n val chain: Array<MutableEntry<K, V>> = chainOrEntry\n val entry = chain.findEntryInChain(key)\n if (entry != null) {\n return entry.setValue(value)\n }\n chain.asDynamic().push(SimpleEntry(key, value))\n }\n }\n size++\n// structureChanged(host)\n return null\n }\n\n override fun remove(key: K): V? {\n val hashCode = equality.getHashCode(key)\n val chainOrEntry = getChainOrEntryOrNull(hashCode) ?: return null\n if (chainOrEntry !is Array<*>) {\n val entry: MutableEntry<K, V> = chainOrEntry\n if (equality.equals(entry.key, key)) {\n jsDeleteProperty(backingMap, hashCode)\n size--\n return entry.value\n } else {\n return null\n }\n } else {\n val chain: Array<MutableEntry<K, V>> = chainOrEntry\n for (index in chain.indices) {\n val entry = chain[index]\n if (equality.equals(key, entry.key)) {\n if (chain.size == 1) {\n chain.asDynamic().length = 0\n // remove the whole array\n jsDeleteProperty(backingMap, hashCode)\n } else {\n // splice out the entry we're removing\n chain.asDynamic().splice(index, 1)\n }\n size--\n// structureChanged(host)\n return entry.value\n }\n }\n }\n return null\n }\n\n override fun clear() {\n backingMap = createJsMap()\n size = 0\n }\n\n override fun contains(key: K): Boolean = getEntry(key) != null\n\n override fun get(key: K): V? = getEntry(key)?.value\n\n private fun getEntry(key: K): MutableEntry<K, V>? {\n val chainOrEntry = getChainOrEntryOrNull(equality.getHashCode(key)) ?: return null\n if (chainOrEntry !is Array<*>) {\n val entry: MutableEntry<K, V> = chainOrEntry\n if (equality.equals(entry.key, key)) {\n return entry\n } else {\n return null\n }\n } else {\n val chain: Array<MutableEntry<K, V>> = chainOrEntry\n return chain.findEntryInChain(key)\n }\n }\n\n private fun Array<MutableEntry<K, V>>.findEntryInChain(key: K): MutableEntry<K, V>? =\n firstOrNull { entry -> equality.equals(entry.key, key) }\n\n override fun iterator(): MutableIterator<MutableEntry<K, V>> {\n\n return object : MutableIterator<MutableEntry<K, V>> {\n var state = -1 // -1 not ready, 0 - ready, 1 - done\n\n val keys: Array<String> = js(\"Object\").keys(backingMap)\n var keyIndex = -1\n\n var chainOrEntry: dynamic = null\n var isChain = false\n var itemIndex = -1\n var lastEntry: MutableEntry<K, V>? = null\n\n private fun computeNext(): Int {\n if (chainOrEntry != null && isChain) {\n val chainSize: Int = chainOrEntry.unsafeCast<Array<MutableEntry<K, V>>>().size\n if (++itemIndex < chainSize)\n return 0\n }\n\n if (++keyIndex < keys.size) {\n chainOrEntry = backingMap[keys[keyIndex]]\n isChain = chainOrEntry is Array<*>\n itemIndex = 0\n return 0\n } else {\n chainOrEntry = null\n return 1\n }\n }\n\n override fun hasNext(): Boolean {\n if (state == -1)\n state = computeNext()\n return state == 0\n }\n\n override fun next(): MutableEntry<K, V> {\n if (!hasNext()) throw NoSuchElementException()\n val lastEntry = if (isChain) {\n chainOrEntry.unsafeCast<Array<MutableEntry<K, V>>>()[itemIndex]\n } else {\n chainOrEntry.unsafeCast<MutableEntry<K, V>>()\n }\n this.lastEntry = lastEntry\n state = -1\n return lastEntry\n }\n\n override fun remove() {\n checkNotNull(lastEntry)\n this@InternalHashCodeMap.remove(lastEntry!!.key)\n lastEntry = null\n // the chain being iterated just got modified by InternalHashCodeMap.remove\n itemIndex--\n }\n }\n }\n\n private fun getChainOrEntryOrNull(hashCode: Int): dynamic {\n val chainOrEntry = backingMap[hashCode]\n return if (chainOrEntry === undefined) null else chainOrEntry\n }\n\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n/**\n * The common interface of [InternalStringMap] and [InternalHashCodeMap].\n */\ninternal interface InternalMap<K, V> : MutableIterable<MutableMap.MutableEntry<K, V>> {\n val equality: EqualityComparator\n val size: Int\n

operator fun contains(key: K): Boolean\n   operator fun get(key: K): V?\n\n   fun put(key: K, value: V): V?\n   fun remove(key: K): V?\n   fun clear(): Unit\n\n   fun createJsMap(): dynamic {\n   val result = js(\"Object.create(null)\")\n   // force to switch object representation to dictionary mode\n   result[\"foo\"] = 1\n   jsDeleteProperty(result, \"foo\")\n   return result\n   }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n/*\n * Based on GWT InternalStringMap\n * Copyright 2008 Google Inc.\n */\npackage kotlin.collections\n\nimport kotlin.collections.MutableMap.MutableEntry\n\n/**\n * A simple wrapper around JavaScript Map for key type is string.\n *\n * Though this map is instantiated only with K=String, the K type is not fixed to String statically,\n * because we want to have it erased to Any? in order not to generate type-safe override bridges for\n * [get], [contains], [remove] etc, if they ever are generated.\n */\ninternal class InternalStringMap<K, V>(override val equality: EqualityComparator) : InternalMap<K, V> {\n\n   private var backingMap: dynamic = createJsMap()\n   override var size: Int = 0\n   private set\n\n//   /**\n//   * A mod count to track 'value' replacements in map to ensure that the 'value' that we have in the\n//   * iterator entry is guaranteed to be still correct.\n//   * This is to optimize for the common scenario where the values are not modified during\n//   * iterations where the entries are never stale.\n//   */\n//   private var valueMod: Int = 0\n\n   override operator fun contains(key: K): Boolean {\n   if (key !is String) return false\n   return backingMap[key] !== undefined\n   }\n\n   override operator fun get(key: K): V? {\n   if (key !is String) return null\n   val value = backingMap[key]\n   return if (value !== undefined) value.unsafeCast<V>() else null\n   }\n\n\n   override fun put(key: K, value: V): V? {\n   require(key is String)\n   val oldValue = backingMap[key]\n   backingMap[key] = value\n\n   if (oldValue === undefined) {\n   size++\n//   structureChanged(host)\n   return null\n   } else {\n//   valueMod++\n   return oldValue.unsafeCast<V>()\n   }\n   }\n\n   override fun remove(key: K): V? {\n   if (key !is String) return null\n   val value = backingMap[key]\n   if (value !== undefined) {\n   jsDeleteProperty(backingMap, key)\n   size--\n//   structureChanged(host)\n   return value.unsafeCast<V>()\n   } else {\n//   valueMod++\n   return null\n   }\n   }\n\n\n   override fun clear() {\n   backingMap = createJsMap()\n   size = 0\n   }\n\n\n   override fun iterator(): MutableIterator<MutableEntry<K, V>> {\n   return object : MutableIterator<MutableEntry<K, V>> {\n   private val keys: Array<String> = js(\"Object\").keys(backingMap)\n   private val iterator = keys.iterator()\n   private var lastKey: String? = null\n\n   override fun hasNext(): Boolean = iterator.hasNext()\n\n   override fun next(): MutableEntry<K, V> {\n   val key = iterator.next()\n   lastKey = key\n   @Suppress(\"UNCHECKED_CAST\")\n   return newMapEntry(key as K)\n   }\n\n   override fun remove() {\n   @Suppress(\"UNCHECKED_CAST\")\n   this@InternalStringMap.remove(checkNotNull(lastKey) as K)\n   }\n   }\n   }\n\n   private fun newMapEntry(key: K): MutableEntry<K, V> = object : MutableEntry<K, V> {\n   override val key: K get() = key\n   override val value: V get() = this@InternalStringMap[key].unsafeCast<V>()\n\n   override fun setValue(newValue: V): V = this@InternalStringMap.put(key, newValue).unsafeCast<V>()\n\n   override fun hashCode(): Int = AbstractMap.entryHashCode(this)\n   override fun toString(): String = AbstractMap.entryToString(this)\n   override fun equals(other: Any?): Boolean = AbstractMap.entryEquals(this, other)\n   }\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n/*\n * Based on GWT LinkedHashMap\n * Copyright 2008 Google Inc.\n */\npackage kotlin.collections\n\nimport kotlin.collections.MutableMap.MutableEntry\n\n/**\n * Hash table based implementation of the [MutableMap] interface, which additionally preserves the insertion order\n * of entries during the iteration.\n *\n * The insertion order is preserved by maintaining a doubly-linked list of all of its entries.\n */\npublic actual open class LinkedHashMap<K, V> : HashMap<K, V>, MutableMap<K, V> {\n\n   /**\n   * The entry we use includes next/prev pointers for a doubly-linked circular\n   * list with a head node. This reduces the special cases we have to deal with\n   * in the list operations.\n\n   * Note that we duplicate the key from the underlying hash map so we can find\n   * the eldest entry. The alternative would have been to modify HashMap so

```
more\n     * of the code was directly usable here, but this would have added some\n     * overhead to HashMap, or to reimplement most of the HashMap code here with\n     * small modifications. Paying a small storage cost only if you use\n     * LinkedHashMap and minimizing code size seemed like a better tradeoff\n     */\n    private inner class ChainEntry<K, V>(key: K, value: V) : AbstractMutableMap.SimpleEntry<K, V>(key, value) {\n        internal var next: ChainEntry<K, V>? = null\n        internal var prev: ChainEntry<K, V>? = null\n\n        override fun setValue(newValue: V): V {\n            this@LinkedHashMap.checkIsMutable()\n            return super.setValue(newValue)\n        }\n    }\n\n    private inner class EntrySet : AbstractEntrySet<MutableEntry<K, V>, K, V>() {\n\n        private inner class EntryIterator : MutableIterator<MutableEntry<K, V>> {\n            // The last entry that was returned from this iterator.\n            private var last: ChainEntry<K, V>? = null\n\n            // The next entry to return from this iterator.\n            private var next: ChainEntry<K, V>? = null\n\n            init {\n                next = head\n//                recordLastKnownStructure(map, this)\n            }\n\n            override fun hasNext(): Boolean {\n                return next !== null\n            }\n\n            override fun next(): MutableEntry<K, V> {\n//                checkStructuralChange(map, this)\n                if (!hasNext()) throw NoSuchElementException()\n\n                val current = next!!\n                last = current\n                next = current.next.takeIf { it !== head }\n                return current\n            }\n\n            override fun remove() {\n                check(last != null)\n                this@EntrySet.checkIsMutable()\n//                checkStructuralChange(map, this)\n\n                last!!.remove()\n                map.remove(last!!.key)\n//                recordLastKnownStructure(map, this)\n                last = null\n            }\n        }\n\n        override fun add(element: MutableEntry<K, V>): Boolean = throw UnsupportedOperationException(\"Add is not supported on entries\")\n        override fun clear() {\n            this@LinkedHashMap.clear()\n        }\n\n        override fun containsEntry(element: Map.Entry<K, V>): Boolean = this@LinkedHashMap.containsEntry(element)\n\n        override operator fun iterator(): MutableIterator<MutableEntry<K, V>> = EntryIterator()\n\n        override fun removeEntry(element: Map.Entry<K, V>): Boolean {\n            checkIsMutable()\n            if (contains(element)) {\n                this@LinkedHashMap.remove(element.key)\n                return true\n            }\n            return false\n        }\n\n        override val size: Int get() = this@LinkedHashMap.size\n\n        override fun checkIsMutable(): Unit = this@LinkedHashMap.checkIsMutable()\n    }\n\n\n    /*\n     * The head of the insert order chain, which is a doubly-linked circular\n     * list.\n     *\n     * The most recently inserted node is at the end of the chain, ie.\n     * chain.prev.\n     */\n    private var head: ChainEntry<K, V>? = null\n\n    /**\n     * Add this node to the end of the chain.\n     */\n    private fun ChainEntry<K, V>.addToEnd() {\n        // This entry is not in the list.\n        check(next == null && prev == null)\n\n        val _head = head\n        if (_head == null) {\n            head = this\n            next = this\n            prev = this\n        } else {\n            // Chain is valid.\n            val _tail = checkNotNull(_head.prev)\n            // Update me.\n            prev = _tail\n            next = _head\n            // Update my new siblings: current head and old tail\n            _head.prev = this\n            _tail.next = this\n        }\n    }\n\n    /**\n     * Remove this node from the chain it is a part of.\n     */\n    private fun ChainEntry<K, V>.remove() {\n        if (this.next === this) {\n            // if this is single element, remove head\n            head = null\n        } else {\n            if (head === this) {\n                // if this is first element, move head to next\n                head = next\n            }\n            next!!.prev = prev\n            prev!!.next = next\n        }\n        next = null\n        prev = null\n    }\n\n    /*\n     * The hashmap that keeps track of our entries and the chain. Note that we\n     * duplicate the key here to eliminate changes to HashMap and minimize the\n     * code here, at the expense of additional space.\n     */\n    private val map: HashMap<K, ChainEntry<K, V>>\n\n    private var isReadOnly: Boolean = false\n\n    /**\n     * Constructs an empty [LinkedHashMap] instance.\n     */\n    actual constructor() : super() {\n        map = HashMap<K, ChainEntry<K, V>>()\n    }\n\n    internal constructor(backingMap: HashMap<K, Any>) : super() {\n        @Suppress(\"UNCHECKED_CAST\") // expected to work due to erasure\n        map = backingMap as HashMap<K, ChainEntry<K, V>>\n    }\n\n    /**\n     * Constructs an empty [LinkedHashMap] instance.\n     *\n     * @param  initialCapacity the initial capacity (ignored)\n     * @param  loadFactor      the load factor (ignored)\n     *\n     * @throws IllegalArgumentException if the initial capacity or load factor are negative\n     */\n    actual constructor(initialCapacity: Int, loadFactor: Float) : super(initialCapacity, loadFactor) {\n        map = HashMap<K, ChainEntry<K, V>>()\n    }\n\n    actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)\n\n    /**\n     * Constructs an instance of
```

[LinkedHashMap] filled with the contents of the specified [original] map.\n   */\n   actual constructor(original: Map<out K, V>) {\n   map = HashMap<K, ChainEntry<K, V>>()\n   this.putAll(original)\n   }\n\n   @PublishedApi\n   internal fun build(): Map<K, V> {\n   checkIsMutable()\n   isReadOnly = true\n   return this\n   }\n\n   actual override fun clear() {\n   checkIsMutable()\n   map.clear()\n   head = null\n   }\n\n\n//   override fun clone(): Any {\n//   return LinkedHashMap(this)\n//   }\n\n\n   actual override fun containsKey(key: K): Boolean = map.containsKey(key)\n\n   actual override fun containsValue(value: V): Boolean {\n   var node: ChainEntry<K, V> = head ?: return false\n   do {\n   if (node.value == value) {\n   return true\n   }\n   node = node.next!!\n   } while (node !== head)\n   return false\n   }\n\n\n   internal override fun createEntrySet(): MutableSet<MutableMap.MutableEntry<K, V>> = EntrySet()\n\n   actual override operator fun get(key: K): V? = map.get(key)?.value\n\n   actual override fun put(key: K, value: V): V? {\n   checkIsMutable()\n\n   val old = map.get(key)\n   if (old == null) {\n   val newEntry = ChainEntry(key, value)\n   map.put(key, newEntry)\n   newEntry.addToEnd()\n   return null\n   } else {\n   return old.setValue(value)\n   }\n   }\n\n   actual override fun remove(key: K): V? {\n   checkIsMutable()\n\n   val entry = map.remove(key)\n   if (entry != null) {\n   entry.remove()\n   return entry.value\n   }\n   return null\n   }\n\n   actual override val size: Int get() = map.size\n\n   internal override fun checkIsMutable() {\n   if (isReadOnly) throw UnsupportedOperationException()\n   }\n}\n\n/**\n * Constructs the specialized implementation of [LinkedHashMap] with [String] keys, which stores the keys as properties of\n * JS object without hashing them.\n */\npublic fun <V> linkedStringMapOf(vararg pairs: Pair<String, V>): LinkedHashMap<String, V> {\n   return LinkedHashMap<String, V>(stringMapOf<Any>()).apply { putAll(pairs) }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n/*\n * Based on GWT LinkedHashSet\n * Copyright 2008 Google Inc.\n */\n\npackage kotlin.collections\n\n/**\n * The implementation of the [MutableSet] interface, backed by a [LinkedHashMap] instance.\n *\n * This implementation preserves the insertion order of elements during the iteration.\n */\npublic actual open class LinkedHashSet<E> : HashSet<E>, MutableSet<E> {\n\n   internal constructor(map: LinkedHashMap<E, Any>) : super(map)\n\n   /**\n   * Constructs a new empty [LinkedHashSet].\n   */\n   actual constructor() : super(LinkedHashMap<E, Any>())\n\n   /**\n   * Constructs a new [LinkedHashSet] filled with the elements of the specified collection.\n   */\n   actual constructor(elements: Collection<E>) : super(LinkedHashMap<E, Any>()) {\n   addAll(elements)\n   }\n\n   /**\n   * Constructs a new empty [LinkedHashSet].\n   *\n   * @param initialCapacity the initial capacity (ignored)\n   * @param loadFactor   the load factor (ignored)\n   *\n   * @throws IllegalArgumentException if the initial capacity or load factor are negative\n   */\n   actual constructor(initialCapacity: Int, loadFactor: Float) : super(LinkedHashMap<E, Any>(initialCapacity, loadFactor))\n\n   actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)\n\n   @PublishedApi\n   internal fun build(): Set<E> {\n   (map as LinkedHashMap<E, Any>).build()\n   return this\n   }\n\n   internal override fun checkIsMutable(): Unit = map.checkIsMutable()\n\n//   public override fun clone(): Any {\n//   return LinkedHashSet(this)\n//   }\n\n}\n\n/**\n * Creates a new instance of the specialized implementation of [LinkedHashSet] with the specified [String] elements,\n * which elements the keys as properties of JS object without hashing them.\n */\npublic fun linkedStringSetOf(vararg elements: String): LinkedHashSet<String> {\n   return LinkedHashSet(linkedStringMapOf<Any>()).apply { addAll(elements) }\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\nimport kotlin.contracts.*\n\n\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\n@Deprecated(\"Synchronization on any object is not supported in Kotlin/JS\", ReplaceWith(\"run(block)\"))\n@kotlin.internal.InlineOnly\n@Suppress(\"UNUSED_PARAMETER\")\npublic inline fun <R> synchronized(lock: Any, block: () -> R): R {\n   contract {\n   callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n   }\n   return block()\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that

can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.io\n\ninternal abstract class BaseOutput {\n open fun println() {\n print(\"\\n\")\n }\n\n open fun println(message: Any?) {\n print(message)\n println()\n }\n\n abstract fun print(message: Any?)\n\n open fun flush() {}\n}\n\n/** JsName used to make the declaration available outside of module to test it */\n@JsName(\"NodeJsOutput\")\ninternal class NodeJsOutput(val outputStream: dynamic) : BaseOutput() {\n override fun print(message: Any?) {\n // TODO: Using local variable because of bug in block decomposition lowering in IR backend\n val messageString = String(message)\n outputStream.write(messageString)\n }\n}\n\n/** JsName used to make the declaration available outside of module to test it */\n@JsName(\"OutputToConsoleLog\")\ninternal class OutputToConsoleLog : BaseOutput() {\n override fun print(message: Any?) {\n console.log(message)\n }\n\n override fun println(message: Any?) {\n console.log(message)\n }\n\n override fun println() {\n console.log(\"\")\n }\n}\n\n/** JsName used to make the declaration available outside of module to test it and use at try.kotl.in */\n@JsName(\"BufferedOutput\")\ninternal open class BufferedOutput : BaseOutput() {\n var buffer = \"\"\n\n override fun print(message: Any?) {\n buffer += String(message)\n }\n\n override fun flush() {\n buffer = \"\"\n }\n}\n\n/** JsName used to make the declaration available outside of module to test it */\n@JsName(\"BufferedOutputToConsoleLog\")\ninternal class BufferedOutputToConsoleLog : BufferedOutput() {\n override fun print(message: Any?) {\n var s = String(message)\n val i = s.nativeLastIndexOf(\"\\n\", 0)\n if (i >= 0) {\n buffer += s.substring(0, i)\n flush()\n s = s.substring(i + 1)\n }\n buffer += s\n }\n\n override fun flush() {\n console.log(buffer)\n buffer = \"\"\n }\n}\n\n/** JsName used to make the declaration available outside of module to test it and use at try.kotl.in */\n@JsName(\"output\")\ninternal var output = run {\n val isNode: Boolean = js(\"typeof process !== 'undefined' && process.versions && !!process.versions.node\")\n if (isNode) NodeJsOutput(js(\"process.stdout\")) else BufferedOutputToConsoleLog()\n}\n\n@kotlin.internal.InlineOnly\nprivate inline fun String(value: Any?): String = js(\"String\")(value)\n\n/** Prints the line separator to the standard output stream. */\npublic actual fun println() {\n output.println()\n}\n\n/** Prints the given [message] and the line separator to the standard output stream. */\npublic actual fun println(message: Any?) {\n output.println(message)\n}\n\n/** Prints the given [message] to the standard output stream. */\npublic actual fun print(message: Any?) {\n output.print(message)\n}\n\n@SinceKotlin(\"1.6\")\npublic actual fun readln(): String = throw UnsupportedOperationException(\"readln is not supported in Kotlin/JS\")\n\n@SinceKotlin(\"1.6\")\npublic actual fun readlnOrNull(): String? = throw UnsupportedOperationException(\"readlnOrNull is not supported in Kotlin/JS\")","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.coroutines\n\nimport kotlin.coroutines.intrinsics.CoroutineSingletons.*\nimport kotlin.coroutines.intrinsics.COROUTINE_SUSPENDED\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\ninternal actual class SafeContinuation<in T>\ninternal actual constructor(\n private val delegate: Continuation<T>,\n initialResult: Any?\n) : Continuation<T> {\n @PublishedApi\n internal actual constructor(delegate: Continuation<T>) : this(delegate, UNDECIDED)\n\n public actual override val context: CoroutineContext\n get() = delegate.context\n\n private var result: Any? = initialResult\n\n public actual override fun resumeWith(result: Result<T>) {\n val cur = this.result\n when {\n cur === UNDECIDED -> {\n this.result = result.value\n }\n cur === COROUTINE_SUSPENDED -> {\n this.result = RESUMED\n delegate.resumeWith(result)\n }\n else -> throw IllegalStateException(\"Already resumed\")\n }\n }\n\n @PublishedApi\n internal actual fun getOrThrow(): Any? {\n if (result === UNDECIDED) {\n result = COROUTINE_SUSPENDED\n return COROUTINE_SUSPENDED\n }\n val result = this.result\n return when {\n result === RESUMED -> COROUTINE_SUSPENDED // already called continuation, indicate COROUTINE_SUSPENDED upstream\n result is Result.Failure -> throw result.exception\n else -> result // either COROUTINE_SUSPENDED or data\n }\n }\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage

kotlin.coroutines.cancellation\n\n@SinceKotlin(\"1.4\")\npublic actual open class CancellationException : IllegalStateException {\n actual constructor() : super()\n actual constructor(message: String?) : super(message)\n constructor(message: String?, cause: Throwable?) : super(message, cause)\n constructor(cause: Throwable?) : super(cause)\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.coroutines.js.internal\n\nimport kotlin.coroutines.Continuation\nimport kotlin.coroutines.EmptyCoroutineContext\n\n@PublishedApi\n@SinceKotlin(\"1.3\")\ninternal val EmptyContinuation = Continuation<Any?>(EmptyCoroutineContext) { result ->\n result.getOrThrow()\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js\n\n/**\n * Exposes the [Date API](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date) to Kotlin.\n */\n@Suppress(\"NOT_DOCUMENTED\")\npublic external class Date() {\n public constructor(milliseconds: Number)\n\n public constructor(dateString: String)\n\n public constructor(year: Int, month: Int)\n\n public constructor(year: Int, month: Int, day: Int)\n\n public constructor(year: Int, month: Int, day: Int, hour: Int)\n\n public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int)\n\n public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int)\n\n public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int, millisecond: Number)\n\n public fun getDate(): Int\n\n public fun getDay(): Int\n\n public fun getFullYear(): Int\n\n public fun getHours(): Int\n\n public fun getMilliseconds(): Int\n\n public fun getMinutes(): Int\n\n public fun getMonth(): Int\n\n public fun getSeconds(): Int\n\n public fun getTime(): Double\n\n public fun getTimezoneOffset(): Int\n\n public fun getUTCDate(): Int\n\n public fun getUTCDay(): Int\n\n public fun getUTCFullYear(): Int\n\n public fun getUTCHours(): Int\n\n public fun getUTCMilliseconds(): Int\n\n public fun getUTCMinutes(): Int\n\n public fun getUTCMonth(): Int\n\n public fun getUTCSeconds(): Int\n\n public fun toDateString(): String\n\n public fun toISOString(): String\n\n public fun toJSON(): Json\n\n public fun toLocaleDateString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String\n\n public fun toLocaleDateString(locales: String, options: LocaleOptions = definedExternally): String\n\n public fun toLocaleString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String\n\n public fun toLocaleString(locales: String, options: LocaleOptions = definedExternally): String\n\n public fun toLocaleTimeString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String\n\n public fun toLocaleTimeString(locales: String, options: LocaleOptions = definedExternally): String\n\n public fun toTimeString(): String\n\n public fun toUTCString(): String\n\n public companion object {\n public fun now(): Double\n\n public fun parse(dateString: String): Double\n\n public fun UTC(year: Int, month: Int): Double\n\n public fun UTC(year: Int, month: Int, day: Int): Double\n\n public fun UTC(year: Int, month: Int, day: Int, hour: Int): Double\n\n public fun UTC(year: Int, month: Int, day: Int, hour: Int, minute: Int): Double\n\n public fun UTC(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int): Double\n\n public fun UTC(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int, millisecond: Number): Double\n }\n\n public interface LocaleOptions {\n public var localeMatcher: String?\n\n public var timeZone: String?\n\n public var hour12: Boolean?\n\n public var formatMatcher: String?\n\n public var weekday: String?\n\n public var era: String?\n\n public var year: String?\n\n public var month: String?\n\n public var day: String?\n\n public var hour: String?\n\n public var minute: String?\n\n public var second: String?\n\n public var timeZoneName: String?\n }\n}\n\npublic inline fun dateLocaleOptions(init: Date.LocaleOptions.() -> Unit): Date.LocaleOptions {\n val result = js(\"new Object()\").unsafeCast<Date.LocaleOptions>()\n init(result)\n return result\n}","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.dom\n\nimport org.w3c.dom.Document\nimport org.w3c.dom.Element\nimport kotlin.internal.LowPriorityInOverloadResolution\nimport kotlinx.dom.appendElement as

newAppendElement\nimport kotlinx.dom.createElement as newCreateElement\n\n/**\n * Creates a new element with the specified [name].\n *\n * The element is initialized with the specified [init] function.\n */\n@LowPriorityInOverloadResolution\n@Deprecated(\n   message = \"This API is moved to another package, use 'kotlinx.dom.createElement' instead.\",\n   replaceWith = ReplaceWith(\"this.createElement(name, init)\", \"kotlinx.dom.createElement\")\n)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\npublic inline fun Document.createElement(name: String, noinline init: Element.() -> Unit): Element = this.newCreateElement(name, init)\n\n/**\n * Appends a newly created element with the specified [name] to this element.\n *\n * The element is initialized with the specified [init] function.\n */\n@LowPriorityInOverloadResolution\n@Deprecated(\n   message = \"This API is moved to another package, use 'kotlinx.dom.appendElement' instead.\",\n   replaceWith = ReplaceWith(\"this.appendElement(name, init)\", \"kotlinx.dom.appendElement\")\n)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\npublic inline fun Element.appendElement(name: String, noinline init: Element.() -> Unit): Element = this.newAppendElement(name, init)\n\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.dom\n\nimport org.w3c.dom.Element\nimport kotlin.internal.LowPriorityInOverloadResolution\nimport kotlinx.dom.addClass as newAddClass\nimport kotlinx.dom.hasClass as newHasClass\nimport kotlinx.dom.removeClass as newRemoveClass\n\n/** Returns true if the element has the given CSS class style in its 'class' attribute */\n@LowPriorityInOverloadResolution\n@Deprecated(\n   message = \"This API is moved to another package, use 'kotlinx.dom.hasClass' instead.\",\n   replaceWith = ReplaceWith(\"this.hasClass(cssClass)\", \"kotlinx.dom.hasClass\")\n)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\ninline fun Element.hasClass(cssClass: String): Boolean = this.newHasClass(cssClass)\n\n/**\n * Adds CSS class to element. Has no effect if all specified classes are already in class attribute of the element\n *\n * @return true if at least one class has been added\n */\n@LowPriorityInOverloadResolution\n@Deprecated(\n   message = \"This API is moved to another package, use 'kotlinx.dom.addClass' instead.\",\n   replaceWith = ReplaceWith(\"this.addClass(cssClasses)\", \"kotlinx.dom.addClass\")\n)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\ninline fun Element.addClass(vararg cssClasses: String): Boolean = this.newAddClass(*cssClasses)\n\n/**\n * Removes all [cssClasses] from element. Has no effect if all specified classes are missing in class attribute of the element\n *\n * @return true if at least one class has been removed\n */\n@LowPriorityInOverloadResolution\n@Deprecated(\n   message = \"This API is moved to another package, use 'kotlinx.dom.removeClass' instead.\",\n   replaceWith = ReplaceWith(\"this.removeClass(cssClasses)\", \"kotlinx.dom.removeClass\")\n)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\ninline fun Element.removeClass(vararg cssClasses: String): Boolean = this.newRemoveClass(*cssClasses)","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.dom\n\nimport org.w3c.dom.Element\nimport org.w3c.dom.Node\nimport kotlin.internal.LowPriorityInOverloadResolution\nimport kotlinx.dom.isElement as newIsElement\nimport kotlinx.dom.isText as newIsText\n\n/**\n * Gets a value indicating whether this node is a TEXT_NODE or a CDATA_SECTION_NODE.\n */\n@LowPriorityInOverloadResolution\n@Deprecated(\n   message = \"This API is moved to another package, use 'kotlinx.dom.isText' instead.\",\n   replaceWith = ReplaceWith(\"this.isText\", \"kotlinx.dom.isText\")\n)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\npublic val Node.isText: Boolean\n   inline get() = this.newIsText\n\n/**\n * Gets a value indicating whether this node is an [Element].\n */\n@LowPriorityInOverloadResolution\n@Deprecated(\n   message = \"This API is moved to another package, use 'kotlinx.dom.isElement' instead.\",\n   replaceWith = ReplaceWith(\"this.isElement\", \"kotlinx.dom.isElement\")\n)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\npublic val Node.isElement: Boolean\n   inline get() = this.newIsElement\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage org.w3c.dom.events\n\npublic fun

EventListener(handler: (Event) -> Unit): EventListener = EventListenerHandler(handler)\n\nprivate class EventListenerHandler(private val handler: (Event) -> Unit) : EventListener {\n   public override fun handleEvent(event: Event) {\n   handler(event)\n   }\n\n   public override fun toString(): String = \"EventListenerHandler($handler)\"\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage org.w3c.dom\n\npublic external interface ItemArrayLike<out T> {\n   val length: Int\n   fun item(index: Int): T?\n}\n\n/**\n * Returns the view of this `ItemArrayLike<T>` collection as `List<T>`\n */\npublic fun <T> ItemArrayLike<T>.asList(): List<T> = object : AbstractList<T>() {\n   override val size: Int get() = this@asList.length\n\n   override fun get(index: Int): T = when (index) {\n   in 0..lastIndex -> this@asList.item(index).unsafeCast<T>()\n   else -> throw IndexOutOfBoundsException(\"index $index is not in range [0..$lastIndex]\")\n   }\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.dom\n\nimport org.w3c.dom.Element\nimport org.w3c.dom.Node\nimport kotlin.internal.LowPriorityInOverloadResolution\nimport kotlinx.dom.appendText as newAppendText\nimport kotlinx.dom.clear as newClear\n\n/** Removes all the children from this node. */\n@LowPriorityInOverloadResolution\n@Deprecated(\n   message = \"This API is moved to another package, use 'kotlinx.dom.clear' instead.\",\n   replaceWith = ReplaceWith(\"this.clear()\", \"kotlinx.dom.clear\")\n)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\npublic inline fun Node.clear() = this.newClear()\n\n/**\n * Creates text node and append it to the element.\n *\n * @return this element\n */\n@LowPriorityInOverloadResolution\n@Deprecated(\n   message = \"This API is moved to another package, use 'kotlinx.dom.appendText' instead.\",\n   replaceWith = ReplaceWith(\"this.appendText(text)\", \"kotlinx.dom.appendText\")\n)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\ninline fun Element.appendText(text: String): Element = this.newAppendText(text)\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js\n\n/**\n * Reinterprets this value as a value of the [dynamic type](/docs/reference/dynamic-type.html).\n */\n@kotlin.internal.InlineOnly\npublic inline fun Any?.asDynamic(): dynamic = this\n\n/**\n * Reinterprets this value as a value of the specified type [T] without any actual type checking.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Any?.unsafeCast(): @kotlin.internal.NoInfer T = this.asDynamic()\n\n/**\n * Reinterprets this `dynamic` value as a value of the specified type [T] without any actual type checking.\n */\n@kotlin.internal.DynamicExtension\n@JsName(\"unsafeCastDynamic\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> dynamic.unsafeCast(): @kotlin.internal.NoInfer T = this\n\n/**\n * Allows to iterate this `dynamic` object in the following cases:\n * - when it has an `iterator` function,\n * - when it is an array\n * - when it is an instance of [kotlin.collections.Iterable]\n */\n@kotlin.internal.DynamicExtension\npublic operator fun dynamic.iterator(): Iterator<dynamic> {\n   val r: Any? = this\n\n   return when {\n   this[\"iterator\"] != null -> \n   this[\"iterator\"]()\n   isArrayish(r) -> \n   r.unsafeCast<Array<*>>().iterator()\n\n   else -> \n   (r as Iterable<*>).iterator()\n   }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// a package is omitted to get declarations directly under the module\n\n@JsName(\"throwNPE\")\ninternal fun throwNPE(message: String) {\n   throw NullPointerException(message)\n}\n\n@JsName(\"throwCCE\")\ninternal fun throwCCE() {\n   throw ClassCastException(\"Illegal cast\")\n}\n\n@JsName(\"throwISE\")\ninternal fun throwISE(message: String) {\n   throw IllegalStateException(message)\n}\n\n@JsName(\"throwUPAE\")\ninternal fun throwUPAE(propertyName: String) {\n   throw UninitializedPropertyAccessException(\"lateinit property ${propertyName} has not been initialized\")\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\npackage kotlin.collections\n\n/**\n * Groups elements from the [Grouping] source by key and counts elements in each group.\n *\n * @return a [Map] associating the key of each group with the count of elements in the group.\n

*\n * @sample samples.collections.Grouping.groupingByEachCount\n */\n@SinceKotlin(\"1.1\")\npublic actual fun <T, K> Grouping<T, K>.eachCount(): Map<K, Int> =\n   fold(0) { acc, _ -> acc + 1 }\n\n/*\n/**\n * Groups elements from the [Grouping] source by key and sums values provided by the [valueSelector] function for elements in each group.\n *\n * @return a [Map] associating the key of each group with the count of element in the group.\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K> Grouping<T, K>.eachSumOf(valueSelector: (T) -> Int): Map<K, Int> =\n   fold(0) { acc, e -> acc + valueSelector(e) }\n*/","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmName(\"GroupingKt\")\n@file:kotlin.jvm.JvmMultifileClass\n\npackage kotlin.collections\n\n/**\n * Represents a source of elements with a [keyOf] function, which can be applied to each element to get its key.\n *\n * A [Grouping] structure serves as an intermediate step in group-and-fold operations:\n * they group elements by their keys and then fold each group with some aggregating operation.\n *\n * It is created by attaching `keySelector: (T) -> K` function to a source of elements.\n * To get an instance of [Grouping] use one of `groupingBy` extension functions:\n * - [Iterable.groupingBy]\n * - [Sequence.groupingBy]\n * - [Array.groupingBy]\n * - [CharSequence.groupingBy]\n *\n * For the list of group-and-fold operations available, see the [extension functions](#extension-functions) for `Grouping`.\n */\n@SinceKotlin(\"1.1\")\npublic interface Grouping<T, out K> {\n   /** Returns an [Iterator] over the elements of the source of this grouping. */\n   fun sourceIterator(): Iterator<T>\n   /** Extracts the key of an [element]. */\n   fun keyOf(element: T): K\n}\n\n/**\n * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially,\n * passing the previously accumulated value and the current element as arguments, and stores the results in a new map.\n *\n * The key for each element is provided by the [Grouping.keyOf] function.\n *\n * @param operation function is invoked on each element with the following parameters:\n * - `key`: the key of the group this element belongs to;\n * - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;\n * - `element`: the element from the source being aggregated;\n * - `first`: indicates whether it's the first `element` encountered in the group.\n *\n * @return a [Map] associating the key of each group with the result of aggregation of the group elements.\n * @sample samples.collections.Grouping.aggregateByRadix\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K, R> Grouping<T, K>.aggregate(\n   operation: (key: K, accumulator: R?, element: T, first: Boolean) -> R\n): Map<K, R> {\n   return aggregateTo(mutableMapOf<K, R>(), operation)\n}\n\n/**\n * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially,\n * passing the previously accumulated value and the current element as arguments,\n * and stores the results in the given [destination] map.\n *\n * The key for each element is provided by the [Grouping.keyOf] function.\n *\n * @param operation a function that is invoked on each element with the following parameters:\n * - `key`: the key of the group this element belongs to;\n * - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;\n * - `element`: the element from the source being aggregated;\n * - `first`: indicates whether it's the first `element` encountered in the group.\n *\n * If the [destination] map already has a value corresponding to some key,\n * then the elements being aggregated for that key are never considered as `first`.\n *\n * @return the [destination] map associating the key of each group with the result of aggregation of the group elements.\n * @sample samples.collections.Grouping.aggregateByRadixTo\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K, R, M : MutableMap<in K, R>> Grouping<T, K>.aggregateTo(\n   destination: M,\n   operation: (key: K, accumulator: R?, element: T, first: Boolean) -> R\n): M {\n   for (e in this.sourceIterator()) {\n   val key = keyOf(e)\n   val accumulator = destination[key]\n   destination[key] = operation(key, accumulator, e, accumulator == null && !destination.containsKey(key))\n   }\n   return destination\n}\n\n/**\n * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially,\n * passing the previously accumulated value and the current element as arguments, and stores the results in a new map.\n * An initial value of accumulator is provided by [initialValueSelector] function.\n *\n * @param initialValueSelector a function that provides an initial value of accumulator for each group.\n * It's invoked with parameters:\n * - `key`: the key of the group;\n * - `element`: the

first element being encountered in that group.\n *\n * @param operation a function that is invoked on each element with the following parameters:\n * - `key`: the key of the group this element belongs to;\n * - `accumulator`: the current value of the accumulator of the group;\n * - `element`: the element from the source being accumulated.\n *\n * @return a [Map] associating the key of each group with the result of accumulating the group elements.\n * @sample samples.collections.Grouping.foldByEvenLengthWithComputedInitialValue\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K, R> Grouping<T, K>.fold(\n   initialValueSelector: (key: K, element: T) -> R,\n   operation: (key: K, accumulator: R, element: T) -> R\n): Map<K, R> =\n   @Suppress(\"UNCHECKED_CAST\")\n   aggregate { key, acc, e, first -> operation(key, if (first) initialValueSelector(key, e) else acc as R, e) }\n\n/**\n * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially,\n * passing the previously accumulated value and the current element as arguments,\n * and stores the results in the given [destination] map.\n * An initial value of accumulator is provided by [initialValueSelector] function.\n *\n * @param initialValueSelector a function that provides an initial value of accumulator for each group.\n * It's invoked with parameters:\n * - `key`: the key of the group;\n * - `element`: the first element being encountered in that group.\n *\n * If the [destination] map already has a value corresponding to some key, that value is used as an initial value of\n * the accumulator for that group and the [initialValueSelector] function is not called for that group.\n *\n * @param operation a function that is invoked on each element with the following parameters:\n * - `key`: the key of the group this element belongs to;\n * - `accumulator`: the current value of the accumulator of the group;\n * - `element`: the element from the source being accumulated.\n *\n * @return the [destination] map associating the key of each group with the result of accumulating the group elements.\n * @sample samples.collections.Grouping.foldByEvenLengthWithComputedInitialValueTo\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K, R, M : MutableMap<in K, R>> Grouping<T, K>.foldTo(\n   destination: M,\n   initialValueSelector: (key: K, element: T) -> R,\n   operation: (key: K, accumulator: R, element: T) -> R\n): M =\n   @Suppress(\"UNCHECKED_CAST\")\n   aggregateTo(destination) { key, acc, e, first -> operation(key, if (first) initialValueSelector(key, e) else acc as R, e) }\n\n/**\n * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially,\n * passing the previously accumulated value and the current element as arguments, and stores the results in a new map.\n * An initial value of accumulator is the same [initialValue] for each group.\n *\n * @param operation a function that is invoked on each element with the following parameters:\n * - `accumulator`: the current value of the accumulator of the group;\n * - `element`: the element from the source being accumulated.\n *\n * @return a [Map] associating the key of each group with the result of accumulating the group elements.\n * @sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValue\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K, R> Grouping<T, K>.fold(\n   initialValue: R,\n   operation: (accumulator: R, element: T) -> R\n): Map<K, R> =\n   @Suppress(\"UNCHECKED_CAST\")\n   aggregate { _, acc, e, first -> operation(if (first) initialValue else acc as R, e) }\n\n/**\n * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially,\n * passing the previously accumulated value and the current element as arguments,\n * and stores the results in the given [destination] map.\n * An initial value of accumulator is the same [initialValue] for each group.\n *\n * If the [destination] map already has a value corresponding to the key of some group,\n * that value is used as an initial value of the accumulator for that group.\n *\n * @param operation a function that is invoked on each element with the following parameters:\n * - `accumulator`: the current value of the accumulator of the group;\n * - `element`: the element from the source being accumulated.\n *\n * @return the [destination] map associating the key of each group with the result of accumulating the group elements.\n * @sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValueTo\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K, R, M : MutableMap<in K, R>> Grouping<T, K>.foldTo(\n   destination: M,\n   initialValue: R,\n   operation: (accumulator: R, element: T) -> R\n): M =\n   @Suppress(\"UNCHECKED_CAST\")\n   aggregateTo(destination) { _, acc, e, first -> operation(if (first) initialValue else acc as R, e) }\n\n/**\n * Groups elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group\n * sequentially starting from the second element of the group,\n

* passing the previously accumulated value and the current element as arguments,\n * and stores the results in a new map.\n * An initial value of accumulator is the first element of the group.\n *\n * @param operation a function that is invoked on each subsequent element of the group with the following parameters:\n * - `key`: the key of the group this element belongs to;\n * - `accumulator`: the current value of the accumulator of the group;\n * - `element`: the element from the source being accumulated.\n *\n * @return a [Map] associating the key of each group with the result of accumulating the group elements.\n * @sample samples.collections.Grouping.reduceByMaxVowels\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <S, T : S, K> Grouping<T, K>.reduce(\n   operation: (key: K, accumulator: S, element: T) -> S\n): Map<K, S> =\n   aggregate { key, acc, e, first ->\n   @Suppress(\"UNCHECKED_CAST\")\n   if (first) e else operation(key, acc as S, e)\n   }\n\n/**\n * Groups elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group\n * sequentially starting from the second element of the group,\n * passing the previously accumulated value and the current element as arguments,\n * and stores the results in the given [destination] map.\n * An initial value of accumulator is the first element of the group.\n *\n * If the [destination] map already has a value corresponding to the key of some group,\n * that value is used as an initial value of the accumulator for that group and the first element of that group is also\n * subjected to the [operation].\n\n * @param operation a function that is invoked on each subsequent element of the group with the following parameters:\n * - `accumulator`: the current value of the accumulator of the group;\n * - `element`: the element from the source being folded;\n *\n * @return the [destination] map associating the key of each group with the result of accumulating the group elements.\n * @sample samples.collections.Grouping.reduceByMaxVowelsTo\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <S, T : S, K, M : MutableMap<in K, S>> Grouping<T, K>.reduceTo(\n   destination: M,\n   operation: (key: K, accumulator: S, element: T) -> S\n): M =\n   aggregateTo(destination) { key, acc, e, first ->\n   @Suppress(\"UNCHECKED_CAST\")\n   if (first) e else operation(key, acc as S, e)\n   }\n\n/**\n * Groups elements from the [Grouping] source by key and counts elements in each group to the given [destination] map.\n *\n * If the [destination] map already has a value corresponding to the key of some group,\n * that value is used as an initial value of the counter for that group.\n *\n * @return the [destination] map associating the key of each group with the count of elements in the group.\n *\n * @sample samples.collections.Grouping.groupingByEachCount\n */\n@SinceKotlin(\"1.1\")\npublic fun <T, K, M : MutableMap<in K, Int>> Grouping<T, K>.eachCountTo(destination: M): M =\n   foldTo(destination, 0) { acc, _ -> acc + 1 }\n\n/*\n/**\n * Groups elements from the [Grouping] source by key and sums values provided by the [valueSelector] function for elements in each group\n * to the given [destination] map.\n *\n *\n * If the [destination] map already has a value corresponding to the key of some group,\n * that value is used as an initial value of the sum for that group.\n *\n * @return the [destination] map associating the key of each group with the sum of elements in the group.\n */\n@SinceKotlin(\"1.1\")\npublic inline fun <T, K, M : MutableMap<in K, Int>> Grouping<T, K>.eachSumOfTo(destination: M, valueSelector: (T) -> Int): M =\n   foldTo(destination, 0) { acc, e -> acc + valueSelector(e)}\n*/\n\n\n/*\n// TODO: sum by long and by double overloads\n\npublic inline fun <T, K, M : MutableMap<in K, Long>> Grouping<T, K>.sumEachByLongTo(destination: M, valueSelector: (T) -> Long): M =\n   foldTo(destination, 0L) { acc, e -> acc + valueSelector(e)}\n\npublic inline fun <T, K> Grouping<T, K>.sumEachByLong(valueSelector: (T) -> Long): Map<K, Long> =\n   fold(0L) { acc, e -> acc + valueSelector(e)}\n\npublic inline fun <T, K, M : MutableMap<in K, Double>> Grouping<T, K>.sumEachByDoubleTo(destination: M, valueSelector: (T) -> Double): M =\n   foldTo(destination, 0.0) { acc, e -> acc + valueSelector(e)}\n\npublic inline fun <T, K> Grouping<T, K>.sumEachByDouble(valueSelector: (T) -> Double): Map<K, Double> =\n   fold(0.0) { acc, e -> acc + valueSelector(e)}\n*/\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js\n\n@Retention(AnnotationRetention.BINARY)\n@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY)\ninternal annotation class JsNativeImplementation(val implementation: String)\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n

*/\n\npackage kotlin.js\n\n/**\n * An interface for indexing access to a collection of key-value pairs, where type of key is [String] and type of value is [Any?][Any].\n */\npublic external interface Json {\n /**\n * Calls to the function will be translated to indexing operation (square brackets) on the receiver with [propertyName] as the argument.\n *\n * E.g. for next code:\n * ```kotlin\n * fun test(j: Json, p: String) = j[\"prop\"] + j.get(p)\n * ```\n *\n * will be generated:\n * ```js\n * function test(j, p) {\n * return j[\"prop\"] + j[p];\n * }\n * ```\n */\n operator fun get(propertyName: String): Any?\n\n /**\n * Calls of the function will be translated to an assignment of [value] to the receiver indexed (with square brackets/index operation) with [propertyName].\n *\n * E.g. for the following code:\n * ```kotlin\n * fun test(j: Json, p: String, newValue: Any) {\n * j[\"prop\"] = 1\n * j.set(p, newValue)\n * }\n * ```\n *\n * will be generated:\n * ```js\n * function test(j, p, newValue) {\n * j[\"prop\"] = 1;\n * j[p] = newValue;\n * }\n * }\n * ```\n */\n operator fun set(propertyName: String, value: Any?): Unit\n}\n\n/**\n * Returns a simple JavaScript object (as [Json]) using provided key-value pairs as names and values of its properties.\n */\npublic fun json(vararg pairs: Pair<String, Any?>): Json {\n val res: dynamic = js(\"({})\")\n for ((name, value) in pairs) {\n res[name] = value\n }\n return res\n}\n\n/**\n * Adds key-value pairs from [other] to [this].\n * Returns the original receiver.\n */\npublic fun Json.add(other: Json): Json {\n val keys: Array<String> = js(\"Object\").keys(other)\n for (key in keys) {\n if (other.asDynamic().hasOwnProperty(key)) {\n this[key] = other[key];\n }\n }\n return this\n}\n\n/**\n * Exposes the JavaScript [JSON object](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/JSON) to Kotlin.\n */\n@Suppress(\"NOT_DOCUMENTED\")\npublic external object JSON {\n public fun stringify(o: Any?): String\n public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)): String\n public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)? = definedExternally, space: Int): String\n public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)? = definedExternally, space: String): String\n public fun stringify(o: Any?, replacer: Array<String>): String\n public fun stringify(o: Any?, replacer: Array<String>, space: Int): String\n public fun stringify(o: Any?, replacer: Array<String>, space: String): String\n\n public fun <T> parse(text: String): T\n public fun <T> parse(text: String, reviver: ((key: String, value: Any?) -> Any?)): T\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\npackage kotlin.math\n\n\nimport kotlin.internal.InlineOnly\nimport kotlin.js.JsMath as nativeMath\n\n\n// region ================ Double Math ==\n\n/** Computes the sine of the angle [x] given in radians.\n *\n * Special cases:\n * - `sin(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun sin(x: Double): Double = nativeMath.sin(x)\n\n/** Computes the cosine of the angle [x] given in radians.\n *\n * Special cases:\n * - `cos(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun cos(x: Double): Double = nativeMath.cos(x)\n\n/** Computes the tangent of the angle [x] given in radians.\n *\n * Special cases:\n * - `tan(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun tan(x: Double): Double = nativeMath.tan(x)\n\n/**\n * Computes the arc sine of the value [x];\n * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.\n *\n * Special cases:\n * - `asin(x)` is `NaN`, when `abs(x) > 1` or x is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun asin(x: Double): Double = nativeMath.asin(x)\n\n/**\n * Computes the arc cosine of the value [x];\n * the returned value is an angle in the range from `0.0` to `PI` radians.\n *\n * Special cases:\n * - `acos(x)` is `NaN`, when `abs(x) > 1` or x is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun acos(x: Double): Double = nativeMath.acos(x)\n\n/**\n * Computes the arc tangent of the value [x];\n * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.\n *\n * Special cases:\n * - `atan(NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun atan(x: Double): Double = nativeMath.atan(x)\n\n/**\n * Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond\n * to the rectangular coordinates `(x, y)` by computing the arc tangent of the value [y] / [x];\n * the returned value is an angle in the range from `-PI` to `PI` radians.\n *\n * Special cases:\n * - `atan2(0.0, 0.0)` is `0.0`\n * - `atan2(0.0, x)` is

`0.0` for `x > 0` and `PI` for `x < 0`\n * - `atan2(-0.0, x)` is `-0.0` for 'x > 0` and `-PI` for `x < 0`\n *  - `atan2(y, +Inf)` is `0.0` for `0 < y < +Inf` and `-0.0` for '-Inf < y < 0`\n * - `atan2(y, -Inf)` is `PI` for `0 < y < +Inf` and `-PI` for `-Inf < y < 0`\n * - `atan2(y, 0.0)` is `PI/2` for `y > 0` and `-PI/2` for `y < 0`\n * - `atan2(+Inf, x)` is `PI/2` for finite `x`y\n * - `atan2(-Inf, x)` is `-PI/2` for finite `x`\n * - `atan2(NaN, x)` and `atan2(y, NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun atan2(y: Double, x: Double): Double = nativeMath.atan2(y, x)\n\n/**\n * Computes the hyperbolic sine of the value [x].\n *\n * Special cases:\n * - `sinh(NaN)` is `NaN`\n * - `sinh(+Inf)` is `+Inf`\n * - `sinh(-Inf)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun sinh(x: Double): Double = nativeMath.sinh(x)\n\n/**\n * Computes the hyperbolic cosine of the value [x].\n *\n * Special cases:\n * - `cosh(NaN)` is `NaN`\n * - `cosh(+Inf|-Inf)` is `+Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun cosh(x: Double): Double = nativeMath.cosh(x)\n\n/**\n * Computes the hyperbolic tangent of the value [x].\n *\n * Special cases:\n * - `tanh(NaN)` is `NaN`\n * - `tanh(+Inf)` is `1.0`\n * - `tanh(-Inf)` is `-1.0`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun tanh(x: Double): Double = nativeMath.tanh(x)\n\n/**\n * Computes the inverse hyperbolic sine of the value [x].\n *\n * The returned value is `y` such that `sinh(y) == x`.\n *\n * Special cases:\n * - `asinh(NaN)` is `NaN`\n * - `asinh(+Inf)` is `+Inf`\n * - `asinh(-Inf)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun asinh(x: Double): Double = nativeMath.asinh(x)\n\n/**\n * Computes the inverse hyperbolic cosine of the value [x].\n *\n * The returned value is positive `y` such that `cosh(y) == x`.\n *\n * Special cases:\n * - `acosh(NaN)` is `NaN`\n * - `acosh(x)` is `NaN` when `x < 1`\n * - `acosh(+Inf)` is `+Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun acosh(x: Double): Double = nativeMath.acosh(x)\n\n/**\n * Computes the inverse hyperbolic tangent of the value [x].\n *\n * The returned value is `y` such that `tanh(y) == x`.\n *\n * Special cases:\n * - `tanh(NaN)` is `NaN`\n * - `tanh(x)` is `NaN` when `x > 1` or `x < -1`\n * - `tanh(1.0)` is `+Inf`\n * - `tanh(-1.0)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun atanh(x: Double): Double = nativeMath.atanh(x)\n\n/**\n * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.\n *\n * Special cases:\n * - returns `+Inf` if any of arguments is infinite\n * - returns `NaN` if any of arguments is `NaN` and the other is not infinite\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun hypot(x: Double, y: Double): Double = nativeMath.hypot(x, y)\n\n/**\n * Computes the positive square root of the value [x].\n *\n * Special cases:\n * - `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun sqrt(x: Double): Double = nativeMath.sqrt(x)\n\n/**\n * Computes Euler's number `e` raised to the power of the value [x].\n *\n * Special cases:\n * - `exp(NaN)` is `NaN`\n * - `exp(+Inf)` is `+Inf`\n * - `exp(-Inf)` is `0.0`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun exp(x: Double): Double = nativeMath.exp(x)\n\n/**\n * Computes `exp(x) - 1`.\n *\n * This function can be implemented to produce more precise result for [x] near zero.\n *\n * Special cases:\n * - `expm1(NaN)` is `NaN`\n * - `expm1(+Inf)` is `+Inf`\n * - `expm1(-Inf)` is `-1.0`\n *\n * @see [exp] function.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun expm1(x: Double): Double = nativeMath.expm1(x)\n\n/**\n * Computes the logarithm of the value [x] to the given [base].\n *\n * Special cases:\n * - `log(x, b)` is `NaN` if either `x` or `b` are `NaN`\n * - `log(x, b)` is `NaN` when `x < 0` or `b <= 0` or `b == 1.0`\n * - `log(+Inf, +Inf)` is `NaN`\n * - `log(+Inf, b)` is `+Inf` for `b > 1` and `-Inf` for `b < 1`\n * - `log(0.0, b)` is `-Inf` for `b > 1` and `+Inf` for `b > 1`\n *\n * See also logarithm functions for common fixed bases: [ln], [log10] and [log2].\n */\n@SinceKotlin(\"1.2\")\npublic actual fun log(x: Double, base: Double): Double {\n   if (base <= 0.0 || base == 1.0) return Double.NaN\n   return nativeMath.log(x) / nativeMath.log(base)\n}\n\n/**\n * Computes the natural logarithm (base `E`) of the value [x].\n *\n * Special cases:\n * - `ln(NaN)` is `NaN`\n * - `ln(x)` is `NaN` when `x < 0.0`\n * - `ln(+Inf)` is `+Inf`\n * - `ln(0.0)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun ln(x: Double): Double = nativeMath.log(x)\n\n/**\n * Computes the common logarithm (base 10) of the value [x].\n *\n * @see [ln] function for special cases.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun log10(x: Double): Double = nativeMath.log10(x)\n\n/**\n * Computes the binary logarithm (base 2) of the value [x].\n *\n * @see [ln] function for special cases.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun log2(x: Double): Double =

```
nativeMath.log2(x)\n\n/**\n * Computes `ln(x + 1)`.\n *\n * This function can be implemented to produce more precise result for [x] near zero.\n *\n * Special cases:\n *   - `ln1p(NaN)` is `NaN`\n *   - `ln1p(x)` is `NaN` where `x < -1.0`\n *   - `ln1p(-1.0)` is `-Inf`\n *   - `ln1p(+Inf)` is `+Inf`\n *\n * @see [ln] function\n * @see [expm1] function\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun ln1p(x: Double): Double = nativeMath.log1p(x)\n\n/**\n * Rounds the given value [x] to an integer towards positive infinity.\n\n * @return the smallest double value that is greater than or equal to the given value [x] and is a mathematical integer.\n *\n * Special cases:\n *   - `ceil(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun ceil(x: Double): Double = nativeMath.ceil(x)\n\n/**\n * Rounds the given value [x] to an integer towards negative infinity.\n\n * @return the largest double value that is smaller than or equal to the given value [x] and is a mathematical integer.\n *\n * Special cases:\n *   - `floor(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun floor(x: Double): Double = nativeMath.floor(x)\n\n/**\n * Rounds the given value [x] to an integer towards zero.\n *\n * @return the value [x] having its fractional part truncated.\n *\n * Special cases:\n *   - `truncate(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun truncate(x: Double): Double = nativeMath.trunc(x)\n\n/**\n * Rounds the given value [x] towards the closest integer with ties rounded towards even integer.\n *\n * Special cases:\n *   - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\npublic actual fun round(x: Double): Double {\n    if (x % 0.5 != 0.0) {\n        return nativeMath.round(x)\n    }\n    val floor = floor(x)\n    return if (floor % 2 == 0.0) floor else ceil(x)\n}\n\n/**\n * Returns the absolute value of the given value [x].\n *\n * Special cases:\n *   - `abs(NaN)` is `NaN`\n *\n * @see absoluteValue extension property for [Double]\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun abs(x: Double): Double = nativeMath.abs(x)\n\n/**\n * Returns the sign of the given value [x]:\n *   - `-1.0` if the value is negative,\n *   - zero if the value is zero,\n *   - `1.0` if the value is positive\n *\n * Special case:\n *   - `sign(NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun sign(x: Double): Double = nativeMath.sign(x)\n\n\n/**\n * Returns the smaller of two values.\n *\n * If either value is `NaN`, then the result is `NaN`.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun min(a: Double, b: Double): Double = nativeMath.min(a, b)\n\n/**\n * Returns the greater of two values.\n *\n * If either value is `NaN`, then the result is `NaN`.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun max(a: Double, b: Double): Double = nativeMath.max(a, b)\n\n// extensions\n\n/**\n * Raises this value to the power [x].\n *\n * Special cases:\n *   - `b.pow(0.0)` is `1.0`\n *   - `b.pow(1.0) == b`\n *   - `b.pow(NaN)` is `NaN`\n *   - `NaN.pow(x)` is `NaN` for `x != 0.0`\n *   - `b.pow(Inf)` is `NaN` for `abs(b) == 1.0`\n *   - `b.pow(x)` is `NaN` for `b < 0` and `x` is finite and not an integer\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun Double.pow(x: Double): Double = nativeMath.pow(this, x)\n\n/**\n * Raises this value to the integer power [n].\n *\n * See the other overload of [pow] for details.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun Double.pow(n: Int): Double = nativeMath.pow(this, n.toDouble())\n\n/**\n * Returns the absolute value of this value.\n *\n * Special cases:\n *   - `NaN.absoluteValue` is `NaN`\n *\n * @see abs function\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline val Double.absoluteValue: Double get() = nativeMath.abs(this)\n\n/**\n * Returns the sign of this value:\n *   - `-1.0` if the value is negative,\n *   - zero if the value is zero,\n *   - `1.0` if the value is positive\n *\n * Special case:\n *   - `NaN.sign` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline val Double.sign: Double get() = nativeMath.sign(this)\n\n/**\n * Returns this value with the sign bit same as of the [sign] value.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun Double.withSign(sign: Int): Double = this.withSign(sign.toDouble())\n\n/**\n * Returns the ulp (unit in the last place) of this value.\n *\n * An ulp is a positive distance between this value and the next nearest [Double] value larger in magnitude.\n *\n * Special Cases:\n *   - `NaN.ulp` is `NaN`\n *   - `x.ulp` is `+Inf` when `x` is `+Inf` or `-Inf`\n *   - `0.0.ulp` is `Double.MIN_VALUE`\n */\n@SinceKotlin(\"1.2\")\npublic actual val Double.ulp: Double get() = when {\n    this < 0 -> (-this).ulp\n    this.isNaN() || this == Double.POSITIVE_INFINITY -> this\n    this == Double.MAX_VALUE -> this - this.nextDown()\n    else -> this.nextUp() - this\n}\n\n/**\n * Returns the [Double]
```

value nearest to this value in direction of positive infinity.\n */\n@SinceKotlin(\"1.2\")\npublic actual fun Double.nextUp(): Double = when {\n   this.isNaN() || this == Double.POSITIVE_INFINITY -> this\n   this == 0.0 -> Double.MIN_VALUE\n   else -> Double.fromBits(this.toRawBits() + if (this > 0) 1 else -1)\n}\n\n/**\n * Returns the [Double] value nearest to this value in direction of negative infinity.\n */\n@SinceKotlin(\"1.2\")\npublic actual fun Double.nextDown(): Double = when {\n   this.isNaN() || this == Double.NEGATIVE_INFINITY -> this\n   this == 0.0 -> -Double.MIN_VALUE\n   else -> Double.fromBits(this.toRawBits() + if (this > 0) -1 else 1)\n}\n\n\n/**\n * Returns the [Double] value nearest to this value in direction from this value towards the value [to].\n *\n * Special cases:\n *   - `x.nextTowards(y)` is `NaN` if either `x` or `y` are `NaN`\n *   - `x.nextTowards(x) == x`\n *\n */\n@SinceKotlin(\"1.2\")\npublic actual fun Double.nextTowards(to: Double): Double = when {\n   this.isNaN() || to.isNaN() -> Double.NaN\n   to == this -> to\n   to > this -> this.nextUp()\n   else /* to < this */ -> this.nextDown()\n}\n\n\n/**\n * Rounds this [Double] value to the nearest integer and converts the result to [Int].\n * Ties are rounded towards positive infinity.\n *\n * Special cases:\n *   - `x.roundToInt() == Int.MAX_VALUE` when `x > Int.MAX_VALUE`\n *   - `x.roundToInt() == Int.MIN_VALUE` when `x < Int.MIN_VALUE`\n *\n * @throws IllegalArgumentException when this value is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic actual fun Double.roundToInt(): Int = when {\n   isNaN() -> throw IllegalArgumentException(\"Cannot round NaN value.\")\n   this > Int.MAX_VALUE -> Int.MAX_VALUE\n   this < Int.MIN_VALUE -> Int.MIN_VALUE\n   else -> nativeMath.round(this).toInt()\n}\n\n/**\n * Rounds this [Double] value to the nearest integer and converts the result to [Long].\n * Ties are rounded towards positive infinity.\n *\n * Special cases:\n *   - `x.roundToLong() == Long.MAX_VALUE` when `x > Long.MAX_VALUE`\n *   - `x.roundToLong() == Long.MIN_VALUE` when `x < Long.MIN_VALUE`\n *\n * @throws IllegalArgumentException when this value is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic actual fun Double.roundToLong(): Long = when {\n   isNaN() -> throw IllegalArgumentException(\"Cannot round NaN value.\")\n   this > Long.MAX_VALUE -> Long.MAX_VALUE\n   this < Long.MIN_VALUE -> Long.MIN_VALUE\n   else -> nativeMath.round(this).toLong()\n}\n\n// endregion\n\n\n\n// region ================ Float Math ======================================\n\n/** Computes the sine of the angle [x] given in radians.\n *\n * Special cases:\n *   - `sin(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun sin(x: Float): Float = nativeMath.sin(x.toDouble()).toFloat()\n\n/** Computes the cosine of the angle [x] given in radians.\n *\n * Special cases:\n *   - `cos(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun cos(x: Float): Float = nativeMath.cos(x.toDouble()).toFloat()\n\n/** Computes the tangent of the angle [x] given in radians.\n *\n * Special cases:\n *   - `tan(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun tan(x: Float): Float = nativeMath.tan(x.toDouble()).toFloat()\n\n/**\n * Computes the arc sine of the value [x];\n * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.\n *\n * Special cases:\n *   - `asin(x)` is `NaN`, when `abs(x) > 1` or x is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun asin(x: Float): Float = nativeMath.asin(x.toDouble()).toFloat()\n\n/**\n * Computes the arc cosine of the value [x];\n * the returned value is an angle in the range from `0.0` to `PI` radians.\n *\n * Special cases:\n *   - `acos(x)` is `NaN`, when `abs(x) > 1` or x is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun acos(x: Float): Float = nativeMath.acos(x.toDouble()).toFloat()\n\n/**\n * Computes the arc tangent of the value [x];\n * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.\n *\n * Special cases:\n *   - `atan(NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun atan(x: Float): Float = nativeMath.atan(x.toDouble()).toFloat()\n\n/**\n * Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond\n * to the rectangular coordinates `(x, y)` by computing the arc tangent of the value [y] / [x];\n * the returned value is an angle in the range from `-PI` to `PI` radians.\n *\n * Special cases:\n *   - `atan2(0.0, 0.0)` is `0.0`\n *   - `atan2(0.0, x)` is `0.0` for `x > 0` and `PI` for `x < 0`\n *   - `atan2(-0.0, x)` is `-0.0` for 'x > 0` and `-PI` for `x < 0`\n *   - `atan2(y, +Inf)` is `0.0` for `0 < y < +Inf` and `-0.0` for '-Inf < y < 0`\n *   - `atan2(y, -Inf)` is `PI` for `0 < y < +Inf` and `-PI` for `-Inf < y < 0`\n *   - `atan2(y, 0.0)` is `PI/2` for `y > 0` and `-PI/2` for `y < 0`\n *   - `atan2(+Inf, x)` is `PI/2` for finite `x`y\n *   - `atan2(-Inf, x)` is `-PI/2` for finite `x`\n *   - `atan2(NaN, x)` and

`atan2(y, NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun atan2(y: Float, x: Float): Float = nativeMath.atan2(y.toDouble(), x.toDouble()).toFloat()\n\n/**\n * Computes the hyperbolic sine of the value [x].\n *\n * Special cases:\n *   - `sinh(NaN)` is `NaN`\n *   - `sinh(+Inf)` is `+Inf`\n *   - `sinh(-Inf)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun sinh(x: Float): Float = nativeMath.sinh(x.toDouble()).toFloat()\n\n/**\n * Computes the hyperbolic cosine of the value [x].\n *\n * Special cases:\n *   - `cosh(NaN)` is `NaN`\n *   - `cosh(+Inf|-Inf)` is `+Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun cosh(x: Float): Float = nativeMath.cosh(x.toDouble()).toFloat()\n\n/**\n * Computes the hyperbolic tangent of the value [x].\n *\n * Special cases:\n *   - `tanh(NaN)` is `NaN`\n *   - `tanh(+Inf)` is `1.0`\n *   - `tanh(-Inf)` is `-1.0`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun tanh(x: Float): Float = nativeMath.tanh(x.toDouble()).toFloat()\n\n/**\n * Computes the inverse hyperbolic sine of the value [x].\n *\n * The returned value is `y` such that `sinh(y) == x`.\n *\n * Special cases:\n *   - `asinh(NaN)` is `NaN`\n *   - `asinh(+Inf)` is `+Inf`\n *   - `asinh(-Inf)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun asinh(x: Float): Float = nativeMath.asinh(x.toDouble()).toFloat()\n\n/**\n * Computes the inverse hyperbolic cosine of the value [x].\n *\n * The returned value is positive `y` such that `cosh(y) == x`.\n *\n * Special cases:\n *   - `acosh(NaN)` is `NaN`\n *   - `acosh(x)` is `NaN` when `x < 1`\n *   - `acosh(+Inf)` is `+Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun acosh(x: Float): Float = nativeMath.acosh(x.toDouble()).toFloat()\n\n/**\n * Computes the inverse hyperbolic tangent of the value [x].\n *\n * The returned value is `y` such that `tanh(y) == x`.\n *\n * Special cases:\n *   - `tanh(NaN)` is `NaN`\n *   - `tanh(x)` is `NaN` when `x > 1` or `x < -1`\n *   - `tanh(1.0)` is `+Inf`\n *   - `tanh(-1.0)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun atanh(x: Float): Float = nativeMath.atanh(x.toDouble()).toFloat()\n\n/**\n * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.\n *\n * Special cases:\n *   - returns `+Inf` if any of arguments is infinite\n *   - returns `NaN` if any of arguments is `NaN` and the other is not infinite\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun hypot(x: Float, y: Float): Float = nativeMath.hypot(x.toDouble(), y.toDouble()).toFloat()\n\n/**\n * Computes the positive square root of the value [x].\n *\n * Special cases:\n *   - `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun sqrt(x: Float): Float = nativeMath.sqrt(x.toDouble()).toFloat()\n\n/**\n * Computes Euler's number `e` raised to the power of the value [x].\n *\n * Special cases:\n *   - `exp(NaN)` is `NaN`\n *   - `exp(+Inf)` is `+Inf`\n *   - `exp(-Inf)` is `0.0`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun exp(x: Float): Float = nativeMath.exp(x.toDouble()).toFloat()\n\n/**\n * Computes `exp(x) - 1`.\n *\n * This function can be implemented to produce more precise result for [x] near zero.\n *\n * Special cases:\n *   - `expm1(NaN)` is `NaN`\n *   - `expm1(+Inf)` is `+Inf`\n *   - `expm1(-Inf)` is `-1.0`\n *\n * @see [exp] function.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun expm1(x: Float): Float = nativeMath.expm1(x.toDouble()).toFloat()\n\n/**\n * Computes the logarithm of the value [x] to the given [base].\n *\n * Special cases:\n *   - `log(x, b)` is `NaN` if either `x` or `b` are `NaN`\n *   - `log(x, b)` is `NaN` when `x < 0` or `b <= 0` or `b == 1.0`\n *   - `log(+Inf, +Inf)` is `NaN`\n *   - `log(+Inf, b)` is `+Inf` for `b > 1` and `-Inf` for `b < 1`\n *   - `log(0.0, b)` is `-Inf` for `b > 1` and `+Inf` for `b > 1`\n *\n * See also logarithm functions for common fixed bases: [ln], [log10] and [log2].\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun log(x: Float, base: Float): Float = log(x.toDouble(), base.toDouble()).toFloat()\n\n/**\n * Computes the natural logarithm (base `E`) of the value [x].\n *\n * Special cases:\n *   - `ln(NaN)` is `NaN`\n *   - `ln(x)` is `NaN` when `x < 0.0`\n *   - `ln(+Inf)` is `+Inf`\n *   - `ln(0.0)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun ln(x: Float): Float = nativeMath.log(x.toDouble()).toFloat()\n\n/**\n * Computes the common logarithm (base 10) of the value [x].\n *\n * @see [ln] function for special cases.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun log10(x: Float): Float = nativeMath.log10(x.toDouble()).toFloat()\n\n/**\n * Computes the binary logarithm (base 2) of the value [x].\n *\n * @see [ln] function for special cases.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun log2(x: Float): Float = nativeMath.log2(x.toDouble()).toFloat()\n\n/**\n * Computes `ln(a + 1)`.\n *\n * This function can be implemented

to produce more precise result for [x] near zero.\n *\n * Special cases:\n *   - `ln1p(NaN)` is `NaN`\n *   - `ln1p(x)` is `NaN` where `x < -1.0`\n *   - `ln1p(-1.0)` is `-Inf`\n *   - `ln1p(+Inf)` is `+Inf`\n *\n * @see [ln] function\n * @see [expm1] function\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun ln1p(x: Float): Float = nativeMath.log1p(x.toDouble()).toFloat()\n\n/**\n * Rounds the given value [x] to an integer towards positive infinity.\n\n * @return the smallest Float value that is greater than or equal to the given value [x] and is a mathematical integer.\n *\n * Special cases:\n *   - `ceil(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun ceil(x: Float): Float = nativeMath.ceil(x.toDouble()).toFloat()\n\n/**\n * Rounds the given value [x] to an integer towards negative infinity.\n\n * @return the largest Float value that is smaller than or equal to the given value [x] and is a mathematical integer.\n *\n * Special cases:\n *   - `floor(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun floor(x: Float): Float = nativeMath.floor(x.toDouble()).toFloat()\n\n/**\n * Rounds the given value [x] to an integer towards zero.\n *\n * @return the value [x] having its fractional part truncated.\n *\n * Special cases:\n *   - `truncate(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun truncate(x: Float): Float = truncate(x.toDouble()).toFloat()\n\n/**\n * Rounds the given value [x] towards the closest integer with ties rounded towards even integer.\n *\n * Special cases:\n *   - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun round(x: Float): Float = round(x.toDouble()).toFloat()\n\n\n/**\n * Returns the absolute value of the given value [x].\n *\n * Special cases:\n *   - `abs(NaN)` is `NaN`\n *\n * @see absoluteValue extension property for [Float]\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun abs(x: Float): Float = nativeMath.abs(x.toDouble()).toFloat()\n\n/**\n * Returns the sign of the given value [x]:\n *   - `-1.0` if the value is negative,\n *   - zero if the value is zero,\n *   - `1.0` if the value is positive\n *\n * Special case:\n *   - `sign(NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun sign(x: Float): Float = nativeMath.sign(x.toDouble()).toFloat()\n\n\n\n/**\n * Returns the smaller of two values.\n *\n * If either value is `NaN`, then the result is `NaN`.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun min(a: Float, b: Float): Float = nativeMath.min(a, b)\n\n/**\n * Returns the greater of two values.\n *\n * If either value is `NaN`, then the result is `NaN`.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun max(a: Float, b: Float): Float = nativeMath.max(a, b)\n\n// extensions\n\n\n/**\n * Raises this value to the power [x].\n *\n * Special cases:\n *   - `b.pow(0.0)` is `1.0`\n *   - `b.pow(1.0) == b`\n *   - `b.pow(NaN)` is `NaN`\n *   - `NaN.pow(x)` is `NaN` for `x != 0.0`\n *   - `b.pow(Inf)` is `NaN` for `abs(b) == 1.0`\n *   - `b.pow(x)` is `NaN` for `b < 0` and `x` is finite and not an integer\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun Float.pow(x: Float): Float = nativeMath.pow(this.toDouble(), x.toDouble()).toFloat()\n\n/**\n * Raises this value to the integer power [n].\n *\n * See the other overload of [pow] for details.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun Float.pow(n: Int): Float = nativeMath.pow(this.toDouble(), n.toDouble()).toFloat()\n\n/**\n * Returns the absolute value of this value.\n *\n * Special cases:\n *   - `NaN.absoluteValue` is `NaN`\n *\n * @see abs function\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline val Float.absoluteValue: Float get() = nativeMath.abs(this.toDouble()).toFloat()\n\n/**\n * Returns the sign of this value:\n *   - `-1.0` if the value is negative,\n *   - zero if the value is zero,\n *   - `1.0` if the value is positive\n *\n * Special case:\n *   - `NaN.sign` is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline val Float.sign: Float get() = nativeMath.sign(this.toDouble()).toFloat()\n\n/**\n * Returns this value with the sign bit same as of the [sign] value.\n *\n * If [sign] is `NaN` the sign of the result is undefined.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun Float.withSign(sign: Float): Float = this.toDouble().withSign(sign.toDouble()).toFloat()\n\n/**\n * Returns this value with the sign bit same as of the [sign] value.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun Float.withSign(sign: Int): Float = this.toDouble().withSign(sign.toDouble()).toFloat()\n\n\n/**\n * Rounds this [Float] value to the nearest integer and converts the result to [Int].\n * Ties are rounded towards positive infinity.\n *\n * Special cases:\n *   - `x.roundToInt() == Int.MAX_VALUE` when `x > Int.MAX_VALUE`\n *   - `x.roundToInt() == Int.MIN_VALUE`

when `x < Int.MIN_VALUE`\n *\n * @throws IllegalArgumentException when this value is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun Float.roundToInt(): Int = toDouble().roundToInt()\n\n/**\n * Rounds this [Float] value to the nearest integer and converts the result to [Long].\n * Ties are rounded towards positive infinity.\n *\n * Special cases:\n *   - `x.roundToLong() == Long.MAX_VALUE` when `x > Long.MAX_VALUE`\n *   - `x.roundToLong() == Long.MIN_VALUE` when `x < Long.MIN_VALUE`\n *\n * @throws IllegalArgumentException when this value is `NaN`\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun Float.roundToLong(): Long = toDouble().roundToLong()\n\n\n// endregion\n\n// region ================ Integer Math ==\n\n\n/**\n * Returns the absolute value of the given value [n].\n *\n * Special cases:\n *   - `abs(Int.MIN_VALUE)` is `Int.MIN_VALUE` due to an overflow\n *\n * @see absoluteValue extension property for [Int]\n */\n// TODO: remove manual 'or' when KT-19290 is fixed\n@SinceKotlin(\"1.2\")\npublic actual fun abs(n: Int): Int = if (n < 0) (-n or 0) else n\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun min(a: Int, b: Int): Int = nativeMath.min(a, b)\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline fun max(a: Int, b: Int): Int = nativeMath.max(a, b)\n\n/**\n * Returns the absolute value of this value.\n *\n * Special cases:\n *   - `Int.MIN_VALUE.absoluteValue` is `Int.MIN_VALUE` due to an overflow\n *\n * @see abs function\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline val Int.absoluteValue: Int get() = abs(this)\n\n/**\n * Returns the sign of this value:\n *   - `-1` if the value is negative,\n *   - `0` if the value is zero,\n *   - `1` if the value is positive\n */\n@SinceKotlin(\"1.2\")\npublic actual val Int.sign: Int get() = when {\n   this < 0 -> -1\n   this > 0 -> 1\n   else -> 0\n}\n\n\n\n/**\n * Returns the absolute value of the given value [n].\n *\n * Special cases:\n *   - `abs(Long.MIN_VALUE)` is `Long.MIN_VALUE` due to an overflow\n *\n * @see absoluteValue extension property for [Long]\n */\n@SinceKotlin(\"1.2\")\npublic actual fun abs(n: Long): Long = if (n < 0) -n else n\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.2\")\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline fun min(a: Long, b: Long): Long = if (a <= b) a else b\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.2\")\n@Suppress(\"NOTHING_TO_INLINE\")\npublic actual inline fun max(a: Long, b: Long): Long = if (a >= b) a else b\n\n/**\n * Returns the absolute value of this value.\n *\n * Special cases:\n *   - `Long.MIN_VALUE.absoluteValue` is `Long.MIN_VALUE` due to an overflow\n *\n * @see abs function\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\npublic actual inline val Long.absoluteValue: Long get() = abs(this)\n\n/**\n * Returns the sign of this value:\n *   - `-1` if the value is negative,\n *   - `0` if the value is zero,\n *   - `1` if the value is positive\n */\n@SinceKotlin(\"1.2\")\npublic actual val Long.sign: Int get() = when {\n   this < 0 -> -1\n   this > 0 -> 1\n   else -> 0\n}\n\n\n// endregion\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\n/**\n * Returns `true` if the specified number is a\n * Not-a-Number (NaN) value, `false` otherwise.\n */\npublic actual fun Double.isNaN(): Boolean = this != this\n\n/**\n * Returns `true` if the specified number is a\n * Not-a-Number (NaN) value, `false` otherwise.\n */\npublic actual fun Float.isNaN(): Boolean = this != this\n\n/**\n * Returns `true` if this value is infinitely large in magnitude.\n */\npublic actual fun Double.isInfinite(): Boolean = this == Double.POSITIVE_INFINITY || this == Double.NEGATIVE_INFINITY\n\n/**\n * Returns `true` if this value is infinitely large in magnitude.\n */\npublic actual fun Float.isInfinite(): Boolean = this == Float.POSITIVE_INFINITY || this == Float.NEGATIVE_INFINITY\n\n/**\n * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).\n */\npublic actual fun Double.isFinite(): Boolean = !isInfinite() && !isNaN()\n\n/**\n * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).\n */\npublic actual fun Float.isFinite(): Boolean = !isInfinite() && !isNaN()\n\n\n/**\n * Counts the number of set bits in the binary representation of this [Int] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Int.countOneBits(): Int {\n   // Hacker's Delight 5-1 algorithm\n   var v = this\n   v = (v and 0x55555555) +

(v.ushr(1) and 0x55555555)\n   v = (v and 0x33333333) + (v.ushr(2) and 0x33333333)\n   v = (v and 0x0F0F0F0F) + (v.ushr(4) and 0x0F0F0F0F)\n   v = (v and 0x00FF00FF) + (v.ushr(8) and 0x00FF00FF)\n   v = (v and 0x0000FFFF) + (v.ushr(16))\n   return v\n}\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic actual inline fun Int.countLeadingZeroBits(): Int = JsMath.clz32(this)\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Int.countTrailingZeroBits(): Int =\n   // Hacker's Delight 5-4 algorithm for expressing countTrailingZeroBits with countLeadingZeroBits\n   Int.SIZE_BITS - (this or -this).inv().countLeadingZeroBits()\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [Int] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Int.takeHighestOneBit(): Int =\n   if (this == 0) 0 else 1.shl(Int.SIZE_BITS - 1 - countLeadingZeroBits())\n\n/**\n * Returns a number having a single bit set in the position of the least significant set bit of this [Int] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Int.takeLowestOneBit(): Int =\n   // Hacker's Delight 2-1 algorithm for isolating rightmost 1-bit\n   this and -this\n\n/**\n * Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n *\n * Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally\n * `number.rotateLeft(n) == number.rotateLeft(n % 32)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Int.rotateLeft(bitCount: Int): Int =\n   shl(bitCount) or ushr(Int.SIZE_BITS - bitCount)\n\n\n/**\n * Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n *\n * Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally\n * `number.rotateRight(n) == number.rotateRight(n % 32)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Int.rotateRight(bitCount: Int): Int =\n   shl(Int.SIZE_BITS - bitCount) or ushr(bitCount)\n\n\n/**\n * Counts the number of set bits in the binary representation of this [Long] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Long.countOneBits(): Int =\n   high.countOneBits() + low.countOneBits()\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Long] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Long.countLeadingZeroBits(): Int =\n   when (val high = this.high) {\n   0 -> Int.SIZE_BITS + low.countLeadingZeroBits()\n   else -> high.countLeadingZeroBits()\n   }\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Long] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Long.countTrailingZeroBits(): Int =\n   when (val low = this.low) {\n   0 -> Int.SIZE_BITS + high.countTrailingZeroBits()\n   else -> low.countTrailingZeroBits()\n   }\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [Long] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Long.takeHighestOneBit(): Long =\n   when (val high = this.high) {\n   0 -> Long(low.takeHighestOneBit(), 0)\n   else -> Long(0, high.takeHighestOneBit())\n   }\n\n/**\n * Returns a number having a single bit set in the position of the least significant set bit of this [Long] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun

Long.takeLowestOneBit(): Long =\n   when (val low = this.low) {\n   0 -> Long(0, high.takeLowestOneBit())\n   else -> Long(low.takeLowestOneBit(), 0)\n   }\n\n/**\n * Rotates the binary representation of this [Long] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n *\n * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n * `number.rotateLeft(n) == number.rotateLeft(n % 64)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual fun Long.rotateLeft(bitCount: Int): Long {\n   if ((bitCount and 31) != 0) {\n   val low = this.low\n   val high = this.high\n   val newLow = low.shl(bitCount) or high.ushr(-bitCount)\n   val newHigh = high.shl(bitCount) or low.ushr(-bitCount)\n   return if ((bitCount and 32) == 0) Long(newLow, newHigh) else Long(newHigh, newLow)\n   } else {\n   return if ((bitCount and 32) == 0) this else Long(high, low)\n   }\n}\n\n\n/**\n * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n *\n * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n * `number.rotateRight(n) == number.rotateRight(n % 64)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic actual inline fun Long.rotateRight(bitCount: Int): Long = rotateLeft(-bitCount)\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js\n\nimport kotlin.internal.LowPriorityInOverloadResolution\n\n/**\n * Exposes the JavaScript [Promise object](https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Global_Objects/Promise) to Kotlin.\n */\n@Suppress(\"NOT_DOCUMENTED\")\npublic open external class Promise<out T>(executor: (resolve: (T) -> Unit, reject: (Throwable) -> Unit) -> Unit) {\n   @LowPriorityInOverloadResolution\n   public open fun <S> then(onFulfilled: ((T) -> S)?): Promise<S>\n\n   @LowPriorityInOverloadResolution\n   public open fun <S> then(onFulfilled: ((T) -> S)?, onRejected: ((Throwable) -> S)?): Promise<S>\n\n   public open fun <S> catch(onRejected: (Throwable) -> S): Promise<S>\n\n   public open fun finally(onFinally: () -> Unit): Promise<T>\n\n   companion object {\n   public fun <S> all(promise: Array<out Promise<S>>): Promise<Array<out S>>\n\n   public fun <S> race(promise: Array<out Promise<S>>): Promise<S>\n\n   public fun reject(e: Throwable): Promise<Nothing>\n\n   public fun <S> resolve(e: S): Promise<S>\n   public fun <S> resolve(e: Promise<S>): Promise<S>\n   }\n}\n\n// It's workaround for KT-19672 since we can fix it properly until KT-11265 isn't fixed.\ninline fun <T, S> Promise<Promise<T>>.then(\n   noinline onFulfilled: ((T) -> S)?\n): Promise<S> {\n   return this.unsafeCast<Promise<T>>().then(onFulfilled)\n}\n\ninline fun <T, S> Promise<Promise<T>>.then(\n   noinline onFulfilled: ((T) -> S)?,\n   noinline onRejected: ((Throwable) -> S)?\n): Promise<S> {\n   return this.unsafeCast<Promise<T>>().then(onFulfilled, onRejected)\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.random\n\nimport kotlin.math.pow\n\ninternal actual fun defaultPlatformRandom(): Random =\n   Random(js(\"(Math.random() * Math.pow(2, 32)) | 0\").unsafeCast<Int>())\n\n\nprivate val INV_2_26: Double = 2.0.pow(-26)\nprivate val INV_2_53: Double = 2.0.pow(-53)\ninternal actual fun doubleFromParts(hi26: Int, low27: Int): Double =\n   hi26 * INV_2_26 + low27 * INV_2_53","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.reflect\n\nimport findAssociatedObject\n\n/**\n * The experimental marker for associated objects API.\n *\n * Any usage of a declaration annotated with `@ExperimentalAssociatedObjects` must be accepted either by\n * annotating that usage with the [OptIn] annotation, e.g. `@OptIn(ExperimentalAssociatedObjects::class)`,\n * or by using the compiler argument `-opt-in=kotlin.reflect.ExperimentalAssociatedObjects`.\n */\n@RequiresOptIn(level =

RequiresOptIn.Level.ERROR)\n@Retention(value = AnnotationRetention.BINARY)\npublic annotation class ExperimentalAssociatedObjects\n\n/**\n * Makes the annotated annotation class an associated object key.\n *\n * An associated object key annotation should have single [KClass] parameter.\n * When applied to a class with reference to an object declaration as an argument, it binds\n * the object to the class, making this binding discoverable at runtime using [findAssociatedObject].\n */\n@ExperimentalAssociatedObjects\n@Retention(AnnotationRetention.BINARY)\n@Target(AnnotationTarget.ANNOTATION_CLASS)\npublic annotation class AssociatedObjectKey\n\n/**\n * If [T] is an @[AssociatedObjectKey]-annotated annotation class and [this] class is annotated with @[T] (`S::class`),\n * returns object `S`.\n *\n * Otherwise returns `null`.\n */\n@ExperimentalAssociatedObjects\npublic inline fun <reified T : Annotation> KClass<*>.findAssociatedObject(): Any? =\n   this.findAssociatedObject(T::class)","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js\n\nimport getKClass\nimport kotlin.reflect.KClass\nimport kotlin.reflect.js.internal.KClassImpl\n\n/**\n * Represents the constructor of a class. Instances of `JsClass` can be passed to JavaScript APIs that expect a constructor reference.\n */\nexternal interface JsClass<T : Any> {\n   /**\n   * Returns the unqualified name of the class represented by this instance.\n   */\n   val name: String\n}\n\n/**\n * Obtains a constructor reference for the given `KClass`.\n */\nval <T : Any> KClass<T>.js: JsClass<T>\n   get() = (this as KClassImpl<T>).jClass\n\n/**\n * Obtains a `KClass` instance for the given constructor reference.\n */\nval <T : Any> JsClass<T>.kotlin: KClass<T>\n   get() = getKClass(this)\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.reflect.js.internal\n\nimport kotlin.reflect.*\n\ninternal abstract class KClassImpl<T : Any>(\n   internal open val jClass: JsClass<T>\n) : KClass<T> {\n\n   override val qualifiedName: String?\n   get() = TODO()\n\n   override fun equals(other: Any?): Boolean {\n   return other is KClassImpl<*> && jClass == other.jClass\n   }\n\n   // TODO: use FQN\n   override fun hashCode(): Int = simpleName?.hashCode() ?: 0\n\n   override fun toString(): String {\n   // TODO: use FQN\n   return \"class $simpleName\"\n   }\n}\n\ninternal class SimpleKClassImpl<T : Any>(jClass: JsClass<T>) : KClassImpl<T>(jClass) {\n   override val simpleName: String? = jClass.asDynamic().`$metadata$`?.simpleName.unsafeCast<String?>()\n\n   override fun isInstance(value: Any?): Boolean {\n   return jsIsType(value, jClass)\n   }\n}\n\ninternal class PrimitiveKClassImpl<T : Any>(\n   jClass: JsClass<T>,\n   private val givenSimpleName: String,\n   private val isInstanceFunction: (Any?) -> Boolean\n) : KClassImpl<T>(jClass) {\n   override fun equals(other: Any?): Boolean {\n   if (other !is PrimitiveKClassImpl<*>) return false\n   return super.equals(other) && givenSimpleName == other.givenSimpleName\n   }\n\n   override val simpleName: String? get() = givenSimpleName\n\n   override fun isInstance(value: Any?): Boolean {\n   return isInstanceFunction(value)\n   }\n}\n\ninternal object NothingKClassImpl : KClassImpl<Nothing>(js(\"Object\")) {\n   override val simpleName: String = \"Nothing\"\n\n   override fun isInstance(value: Any?): Boolean = false\n\n   override val jClass: JsClass<Nothing>\n   get() = throw UnsupportedOperationException(\"There's no native JS class for Nothing type\")\n\n   override fun equals(other: Any?): Boolean = other === this\n\n   override fun hashCode(): Int = 0\n}\n\ninternal class ErrorKClass : KClass<Nothing> {\n   override val simpleName: String? get() = error(\"Unknown simpleName for ErrorKClass\")\n   override val qualifiedName: String? get() = error(\"Unknown qualifiedName for ErrorKClass\")\n\n   override fun isInstance(value: Any?): Boolean = error(\"Can's check isInstance on ErrorKClass\")\n\n   override fun equals(other: Any?): Boolean = other === this\n\n   override fun hashCode(): Int = 0\n}","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.reflect\n\ninternal actual inline val KClass<*>.qualifiedOrSimpleName: String?\n   get() = simpleName","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// a package is omitted to get declarations directly under

the module\n\n// TODO: Remove once JsReflectionAPICallChecker supports more reflection types\n@file:Suppress(\"Unsupported\")\n\nimport kotlin.reflect.*\nimport kotlin.reflect.js.internal.*\n\n@JsName(\"createKType\")\ninternal fun createKType(\n classifier: KClassifier,\n arguments: Array<KTypeProjection>,\n isMarkedNullable: Boolean\n) =\n KTypeImpl(classifier, arguments.asList(), isMarkedNullable)\n\n@JsName(\"createDynamicKType\")\ninternal fun createDynamicKType(): KType = DynamicKType\n\n@JsName(\"markKTypeNullable\")\ninternal fun markKTypeNullable(kType: KType) = KTypeImpl(kType.classifier!!, kType.arguments, true)\n\n@JsName(\"createKTypeParameter\")\ninternal fun createKTypeParameter(\n name: String,\n upperBounds: Array<KType>,\n variance: String\n): KTypeParameter {\n val kVariance = when (variance) {\n \"in\" -> KVariance.IN\n \"out\" -> KVariance.OUT\n else -> KVariance.INVARIANT\n }\n\n return KTypeParameterImpl(name, upperBounds.asList(), kVariance, false)\n}\n\n@JsName(\"getStarKTypeProjection\")\ninternal fun getStarKTypeProjection(): KTypeProjection =\n KTypeProjection.STAR\n\n@JsName(\"createCovariantKTypeProjection\")\ninternal fun createCovariantKTypeProjection(type: KType): KTypeProjection =\n KTypeProjection.covariant(type)\n\n@JsName(\"createInvariantKTypeProjection\")\ninternal fun createInvariantKTypeProjection(type: KType): KTypeProjection =\n KTypeProjection.invariant(type)\n\n@JsName(\"createContravariantKTypeProjection\")\ninternal fun createContravariantKTypeProjection(type: KType): KTypeProjection =\n KTypeProjection.contravariant(type)\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.reflect.js.internal\n\nimport kotlin.reflect.*\n\ninternal class KTypeImpl(\n override val classifier: KClassifier,\n override val arguments: List<KTypeProjection>,\n override val isMarkedNullable: Boolean\n) : KType {\n override fun equals(other: Any?): Boolean =\n other is KTypeImpl &&\n classifier == other.classifier && arguments == other.arguments && isMarkedNullable == other.isMarkedNullable\n\n override fun hashCode(): Int =\n (classifier.hashCode() * 31 + arguments.hashCode()) * 31 + isMarkedNullable.hashCode()\n\n override fun toString(): String {\n val kClass = (classifier as? KClass<*>)\n val classifierName = when {\n kClass == null -> classifier.toString()\n kClass.simpleName != null -> kClass.simpleName\n else -> \"(non-denotable type)\"\n }\n\n val args =\n if (arguments.isEmpty()) \"\"\n else arguments.joinToString(\", \", \"<\", \">\")\n val nullable = if (isMarkedNullable) \"?\" else \"\"\n\n return classifierName + args + nullable\n }\n}\n\ninternal object DynamicKType : KType {\n override val classifier: KClassifier? = null\n override val arguments: List<KTypeProjection> = emptyList()\n override val isMarkedNullable: Boolean = false\n override fun toString(): String = \"dynamic\"\n}\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.reflect.js.internal\n\nimport kotlin.reflect.*\n\ninternal data class KTypeParameterImpl(\n override val name: String,\n override val upperBounds: List<KType>,\n override val variance: KVariance,\n override val isReified: Boolean\n) : KTypeParameter {\n override fun toString(): String = name\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.reflect.js.internal\n\nimport kotlin.js.JsClass\n\n@JsName(\"PrimitiveClasses\")\ninternal object PrimitiveClasses {\n @JsName(\"anyClass\")\n val anyClass = PrimitiveKClassImpl(js(\"Object\").unsafeCast<JsClass<Any>>(), \"Any\", { it is Any })\n\n @JsName(\"numberClass\")\n val numberClass = PrimitiveKClassImpl(js(\"Number\").unsafeCast<JsClass<Number>>(), \"Number\", { it is Number })\n\n @JsName(\"nothingClass\")\n val nothingClass = NothingKClassImpl\n\n @JsName(\"booleanClass\")\n val booleanClass = PrimitiveKClassImpl(js(\"Boolean\").unsafeCast<JsClass<Boolean>>(), \"Boolean\", { it is Boolean })\n\n @JsName(\"byteClass\")\n val byteClass = PrimitiveKClassImpl(js(\"Number\").unsafeCast<JsClass<Byte>>(), \"Byte\", { it is Byte })\n\n

@JsName(\"shortClass\")\n   val shortClass = PrimitiveKClassImpl(js(\"Number\").unsafeCast<JsClass<Short>>(), \"Short\", { it is Short })\n\n   @JsName(\"intClass\")\n   val intClass = PrimitiveKClassImpl(js(\"Number\").unsafeCast<JsClass<Int>>(), \"Int\", { it is Int })\n\n   @JsName(\"floatClass\")\n   val floatClass = PrimitiveKClassImpl(js(\"Number\").unsafeCast<JsClass<Float>>(), \"Float\", { it is Float })\n\n   @JsName(\"doubleClass\")\n   val doubleClass = PrimitiveKClassImpl(js(\"Number\").unsafeCast<JsClass<Double>>(), \"Double\", { it is Double })\n\n   @JsName(\"arrayClass\")\n   val arrayClass = PrimitiveKClassImpl(js(\"Array\").unsafeCast<JsClass<Array<*>>>(), \"Array\", { it is Array<*> })\n\n   @JsName(\"stringClass\")\n   val stringClass = PrimitiveKClassImpl(js(\"String\").unsafeCast<JsClass<String>>(), \"String\", { it is String })\n\n   @JsName(\"throwableClass\")\n   val throwableClass = PrimitiveKClassImpl(js(\"Error\").unsafeCast<JsClass<Throwable>>(), \"Throwable\", { it is Throwable })\n\n   @JsName(\"booleanArrayClass\")\n   val booleanArrayClass = PrimitiveKClassImpl(js(\"Array\").unsafeCast<JsClass<BooleanArray>>(), \"BooleanArray\", { it is BooleanArray })\n\n   @JsName(\"charArrayClass\")\n   val charArrayClass = PrimitiveKClassImpl(js(\"Uint16Array\").unsafeCast<JsClass<CharArray>>(), \"CharArray\", { it is CharArray })\n\n   @JsName(\"byteArrayClass\")\n   val byteArrayClass = PrimitiveKClassImpl(js(\"Int8Array\").unsafeCast<JsClass<ByteArray>>(), \"ByteArray\", { it is ByteArray })\n\n   @JsName(\"shortArrayClass\")\n   val shortArrayClass = PrimitiveKClassImpl(js(\"Int16Array\").unsafeCast<JsClass<ShortArray>>(), \"ShortArray\", { it is ShortArray })\n\n   @JsName(\"intArrayClass\")\n   val intArrayClass = PrimitiveKClassImpl(js(\"Int32Array\").unsafeCast<JsClass<IntArray>>(), \"IntArray\", { it is IntArray })\n\n   @JsName(\"longArrayClass\")\n   val longArrayClass = PrimitiveKClassImpl(js(\"Array\").unsafeCast<JsClass<LongArray>>(), \"LongArray\", { it is LongArray })\n\n   @JsName(\"floatArrayClass\")\n   val floatArrayClass = PrimitiveKClassImpl(js(\"Float32Array\").unsafeCast<JsClass<FloatArray>>(), \"FloatArray\", { it is FloatArray })\n\n   @JsName(\"doubleArrayClass\")\n   val doubleArrayClass = PrimitiveKClassImpl(js(\"Float64Array\").unsafeCast<JsClass<DoubleArray>>(), \"DoubleArray\", { it is DoubleArray })\n\n   @JsName(\"functionClass\")\n   fun functionClass(arity: Int): KClassImpl<Any> {\n   return functionClasses.get(arity) ?: run {\n   val result = PrimitiveKClassImpl(js(\"Function\").unsafeCast<JsClass<Any>>(), \"Function$arity\",\n   { jsTypeOf(it) === \"function\" && it.asDynamic().length === arity })\n   functionClasses.asDynamic()[arity] = result\n   result\n   }\n   }\n}\n\nprivate val functionClasses = arrayOfNulls<KClassImpl<Any>>(0)","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// a package is omitted to get declarations directly under the module\n\nimport kotlin.reflect.*\nimport kotlin.reflect.js.internal.*\n\n@JsName(\"getKClass\")\ninternal fun <T : Any> getKClass(jClass: Any /* JsClass<T> | Array<JsClass<T>> */): KClass<T> {\n   return if (js(\"Array\").isArray(jClass)) {\n   getKClassM(jClass.unsafeCast<Array<JsClass<T>>>())\n   } else {\n   getKClass1(jClass.unsafeCast<JsClass<T>>())\n   }\n}\n\n@JsName(\"getKClassM\")\ninternal fun <T : Any> getKClassM(jClasses: Array<JsClass<T>>): KClass<T> = when (jClasses.size) {\n   1 -> getKClass1(jClasses[0])\n   0 -> NothingKClassImpl.unsafeCast<KClass<T>>()\n   else -> ErrorKClass().unsafeCast<KClass<T>>()\n}\n\n@JsName(\"getKClassFromExpression\")\ninternal fun <T : Any> getKClassFromExpression(e: T): KClass<T> =\n   when (jsTypeOf(e)) {\n   \"string\" -> PrimitiveClasses.stringClass\n   \"number\" -> if (jsBitwiseOr(e, 0).asDynamic() === e) PrimitiveClasses.intClass else PrimitiveClasses.doubleClass\n   \"boolean\" -> PrimitiveClasses.booleanClass\n   \"function\" -> PrimitiveClasses.functionClass(e.asDynamic().length)\n   else -> {\n   when {\n   e is BooleanArray -> PrimitiveClasses.booleanArrayClass\n   e is CharArray -> PrimitiveClasses.charArrayClass\n   e is ByteArray -> PrimitiveClasses.byteArrayClass\n   e is

ShortArray -> PrimitiveClasses.shortArrayClass\n e is IntArray -> PrimitiveClasses.intArrayClass\n e is LongArray -> PrimitiveClasses.longArrayClass\n e is FloatArray -> PrimitiveClasses.floatArrayClass\n e is DoubleArray -> PrimitiveClasses.doubleArrayClass\n e is KClass<*> -> KClass::class\n e is Array<*> -> PrimitiveClasses.arrayClass\n else -> {\n val constructor = js(\"Object\").getPrototypeOf(e).constructor\n when {\n constructor === js(\"Object\") -> PrimitiveClasses.anyClass\n constructor === js(\"Error\") -> PrimitiveClasses.throwableClass\n else -> {\n val jsClass: JsClass<T> = constructor\n getKClass1(jsClass)\n }\n }\n }\n }\n }.unsafeCast<KClass<T>>()\n\n@JsName(\"getKClass1\")\ninternal fun <T : Any> getKClass1(jClass: JsClass<T>): KClass<T> {\n if (jClass === js(\"String\")) return PrimitiveClasses.stringClass.unsafeCast<KClass<T>>()\n\n val metadata = jClass.asDynamic().`$metadata$`\n\n return if (metadata != null) {\n if (metadata.`$kClass$` == null) {\n val kClass = SimpleKClassImpl(jClass)\n metadata.`$kClass$` = kClass\n kClass\n } else {\n metadata.`$kClass$`\n }\n } else {\n SimpleKClassImpl(jClass)\n }\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js\n\n/**\n * Exposes the JavaScript [RegExp object](https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Global_Objects/RegExp) to Kotlin.\n */\n@Suppress(\"NOT_DOCUMENTED\")\npublic external class RegExp(pattern: String, flags: String? = definedExternally) {\n\n public fun test(str: String): Boolean\n\n public fun exec(str: String): RegExpMatch?\n\n public override fun toString(): String\n\n /**\n * The lastIndex is a read/write integer property of regular expressions that specifies the index at which to start the next match.\n */\n public var lastIndex: Int\n\n public val global: Boolean\n public val ignoreCase: Boolean\n public val multiline: Boolean\n}\n\n/**\n * Resets the regular expression so that subsequent [RegExp.test] and [RegExp.exec] calls will match starting with the beginning of the input string.\n */\npublic fun RegExp.reset() {\n lastIndex = 0\n}\n\n// TODO: Inherit from array or introduce asArray() extension\n/**\n * Represents the return value of [RegExp.exec].\n */\n@Suppress(\"NOT_DOCUMENTED\")\npublic external interface RegExpMatch {\n public val index: Int\n public val input: String\n public val length: Int\n}\n\n/**\n * Returns the entire text matched by [RegExp.exec] if the [index] parameter is 0, or the text matched by the capturing parenthesis\n * at the given index.\n */\npublic inline operator fun RegExpMatch.get(index: Int): String? = asDynamic()[index]\n\n/**\n * Converts the result of [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent\n * element is the text matched by each capturing parenthesis.\n */\npublic inline fun RegExpMatch.asArray(): Array<out String?> = unsafeCast<Array<out String?>>()\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.sequences\n\ninternal actual class ConstrainedOnceSequence<T> actual constructor(sequence: Sequence<T>) : Sequence<T> {\n private var sequenceRef: Sequence<T>? = sequence\n\n actual override fun iterator(): Iterator<T> {\n val sequence = sequenceRef ?: throw IllegalStateException(\"This sequence can be consumed only once.\")\n sequenceRef = null\n return sequence.iterator()\n }\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n@SinceKotlin(\"1.5\")\npublic actual enum class CharCategory(internal val value: Int, public actual val code: String) {\n /**\n * General category \"Cn\" in the Unicode specification.\n */\n UNASSIGNED(0, \"Cn\"),\n\n /**\n * General category \"Lu\" in the Unicode specification.\n */\n UPPERCASE_LETTER(1, \"Lu\"),\n\n /**\n * General category \"Ll\" in the Unicode specification.\n */\n LOWERCASE_LETTER(2, \"Ll\"),\n\n /**\n * General category \"Lt\" in the Unicode specification.\n */\n TITLECASE_LETTER(3, \"Lt\"),\n\n /**\n * General category \"Lm\" in the Unicode specification.\n */\n MODIFIER_LETTER(4, \"Lm\"),\n\n /**\n * General category \"Lo\" in the Unicode specification.\n */\n OTHER_LETTER(5, \"Lo\"),\n\n /**\n * General category \"Mn\" in the

Unicode specification.\n */\n NON_SPACING_MARK(6, \"Mn\"),\n\n /**\n * General category \"Me\" in the Unicode specification.\n */\n ENCLOSING_MARK(7, \"Me\"),\n\n /**\n * General category \"Mc\" in the Unicode specification.\n */\n COMBINING_SPACING_MARK(8, \"Mc\"),\n\n /**\n * General category \"Nd\" in the Unicode specification.\n */\n DECIMAL_DIGIT_NUMBER(9, \"Nd\"),\n\n /**\n * General category \"Nl\" in the Unicode specification.\n */\n LETTER_NUMBER(10, \"Nl\"),\n\n /**\n * General category \"No\" in the Unicode specification.\n */\n OTHER_NUMBER(11, \"No\"),\n\n /**\n * General category \"Zs\" in the Unicode specification.\n */\n SPACE_SEPARATOR(12, \"Zs\"),\n\n /**\n * General category \"Zl\" in the Unicode specification.\n */\n LINE_SEPARATOR(13, \"Zl\"),\n\n /**\n * General category \"Zp\" in the Unicode specification.\n */\n PARAGRAPH_SEPARATOR(14, \"Zp\"),\n\n /**\n * General category \"Cc\" in the Unicode specification.\n */\n CONTROL(15, \"Cc\"),\n\n /**\n * General category \"Cf\" in the Unicode specification.\n */\n FORMAT(16, \"Cf\"),\n\n /**\n * General category \"Co\" in the Unicode specification.\n */\n PRIVATE_USE(18, \"Co\"),\n\n /**\n * General category \"Cs\" in the Unicode specification.\n */\n SURROGATE(19, \"Cs\"),\n\n /**\n * General category \"Pd\" in the Unicode specification.\n */\n DASH_PUNCTUATION(20, \"Pd\"),\n\n /**\n * General category \"Ps\" in the Unicode specification.\n */\n START_PUNCTUATION(21, \"Ps\"),\n\n /**\n * General category \"Pe\" in the Unicode specification.\n */\n END_PUNCTUATION(22, \"Pe\"),\n\n /**\n * General category \"Pc\" in the Unicode specification.\n */\n CONNECTOR_PUNCTUATION(23, \"Pc\"),\n\n /**\n * General category \"Po\" in the Unicode specification.\n */\n OTHER_PUNCTUATION(24, \"Po\"),\n\n /**\n * General category \"Sm\" in the Unicode specification.\n */\n MATH_SYMBOL(25, \"Sm\"),\n\n /**\n * General category \"Sc\" in the Unicode specification.\n */\n CURRENCY_SYMBOL(26, \"Sc\"),\n\n /**\n * General category \"Sk\" in the Unicode specification.\n */\n MODIFIER_SYMBOL(27, \"Sk\"),\n\n /**\n * General category \"So\" in the Unicode specification.\n */\n OTHER_SYMBOL(28, \"So\"),\n\n /**\n * General category \"Pi\" in the Unicode specification.\n */\n INITIAL_QUOTE_PUNCTUATION(29, \"Pi\"),\n\n /**\n * General category \"Pf\" in the Unicode specification.\n */\n FINAL_QUOTE_PUNCTUATION(30, \"Pf\");\n\n /**\n * Returns `true` if [char] character belongs to this category.\n */\n public actual operator fun contains(char: Char): Boolean = char.getCategoryValue() == this.value\n\n companion object {\n internal fun valueOf(category: Int): CharCategory =\n when (category) {\n in 0..16 -> values()[category]\n in 18..30 -> values()[category - 1]\n else -> throw IllegalArgumentException(\"Category #$category is not defined.\")\n }\n }\n}\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n/**\n * The exception thrown when a character encoding or decoding error occurs.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic actual open class CharacterCodingException(message: String?) : Exception(message) {\n actual constructor() : this(null)\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n/**\n * A mutable sequence of characters.\n *\n * String builder can be used to efficiently perform multiple string manipulation operations.\n */\npublic actual class StringBuilder actual constructor(content: String) : Appendable, CharSequence {\n /**\n * Constructs an empty string builder with the specified initial [capacity].\n *\n * In Kotlin/JS implementation of StringBuilder the initial capacity has no effect on the further performance of operations.\n */\n actual constructor(capacity: Int) : this() {\n }\n\n /** Constructs a string builder that contains the same characters as the specified [content] char sequence. */\n actual constructor(content: CharSequence) : this(content.toString()) {}\n\n /** Constructs an empty string builder. */\n actual constructor() : this(\"\")\n\n private var string: String = if (content !== undefined) content else \"\"\n\n actual override val length: Int\n get() = string.asDynamic().length\n\n actual override fun get(index: Int): Char =\n string.getOrElse(index) { throw IndexOutOfBoundsException(\"index: $index, length: $length}\") }\n\n actual override fun subSequence(startIndex: Int, endIndex: Int): CharSequence = string.substring(startIndex, endIndex)\n\n actual override fun append(value: Char): StringBuilder {\n string += value\n return this\n }\n\n actual

override fun append(value: CharSequence?): StringBuilder {\n string += value.toString()\n return this\n }\n\n actual override fun append(value: CharSequence?, startIndex: Int, endIndex: Int): StringBuilder =\n this.appendRange(value ?: \"null\", startIndex, endIndex)\n\n /**\n * Reverses the contents of this string builder and returns this instance.\n *\n * Surrogate pairs included in this string builder are treated as single characters.\n * Therefore, the order of the high-low surrogates is never reversed.\n *\n * Note that the reverse operation may produce new surrogate pairs that were unpaired low-surrogates and high-surrogates before the operation.\n * For example, reversing `\"\\uDC00\\uD800\"` produces `\"\\uD800\\uDC00\"` which is a valid surrogate pair.\n */\n actual fun reverse(): StringBuilder {\n var reversed = \"\"\n var index = string.length - 1\n while (index >= 0) {\n val low = string[index--]\n if (low.isLowSurrogate() && index >= 0) {\n val high = string[index--]\n if (high.isHighSurrogate()) {\n reversed = reversed + high + low\n } else {\n reversed = reversed + low + high\n }\n } else {\n reversed += low\n }\n }\n string = reversed\n return this\n }\n\n /**\n * Appends the string representation of the specified object [value] to this string builder and returns this instance.\n *\n * The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method,\n * and then that string was appended to this string builder.\n */\n actual fun append(value: Any?): StringBuilder {\n string += value.toString()\n return this\n }\n\n /**\n * Appends the string representation of the specified boolean [value] to this string builder and returns this instance.\n *\n * The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method,\n * and then that string was appended to this string builder.\n */\n @SinceKotlin(\"1.3\")\n actual fun append(value: Boolean): StringBuilder {\n string += value\n return this\n }\n\n /**\n * Appends characters in the specified character array [value] to this string builder and returns this instance.\n *\n * Characters are appended in order, starting at the index 0.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun append(value: CharArray): StringBuilder {\n string += value.concatToString()\n return this\n }\n\n @Deprecated(\"Provided for binary compatibility.\", level = DeprecationLevel.HIDDEN)\n fun append(value: String): StringBuilder = append(value)\n\n /**\n * Appends the specified string [value] to this string builder and returns this instance.\n *\n * If [value] is `null`, then the four characters `\"null\"` are appended.\n */\n @SinceKotlin(\"1.3\")\n actual fun append(value: String?): StringBuilder {\n this.string += value ?: \"null\"\n return this\n }\n\n /**\n * Returns the current capacity of this string builder.\n *\n * The capacity is the maximum length this string builder can have before an allocation occurs.\n *\n * In Kotlin/JS implementation of StringBuilder the value returned from this method may not indicate the actual size of the backing storage.\n */\n @SinceKotlin(\"1.3\")\n// @ExperimentalStdlibApi\n @Deprecated(\"Obtaining StringBuilder capacity is not supported in JS and common code.\", level = DeprecationLevel.ERROR)\n actual fun capacity(): Int = length\n\n /**\n * Ensures that the capacity of this string builder is at least equal to the specified [minimumCapacity].\n *\n * If the current capacity is less than the [minimumCapacity], a new backing storage is allocated with greater capacity.\n * Otherwise, this method takes no action and simply returns.\n *\n * In Kotlin/JS implementation of StringBuilder the size of the backing storage is not extended to comply the given [minimumCapacity],\n * thus calling this method has no effect on the further performance of operations.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun ensureCapacity(minimumCapacity: Int) {\n }\n\n /**\n * Returns the index within this string builder of the first occurrence of the specified [string].\n *\n * Returns `-1` if the specified [string] does not occur in this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun indexOf(string: String): Int = this.string.asDynamic().indexOf(string)\n\n /**\n * Returns the index within this string builder of the first occurrence of the specified [string],\n * starting at the specified [startIndex].\n *\n * Returns `-1` if the specified [string] does not occur in this string builder starting at the specified [startIndex].\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun indexOf(string: String, startIndex: Int): Int = this.string.asDynamic().indexOf(string, startIndex)\n\n /**\n * Returns the index within this string builder of the last occurrence of the specified [string].\n * The last occurrence of empty string `\"\"` is considered to be at the index equal to `this.length`.\n *\n * Returns `-1` if the specified [string] does not

occur in this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun lastIndexOf(string: String): Int = this.string.asDynamic().lastIndexOf(string)\n\n /**\n * Returns the index within this string builder of the last occurrence of the specified [string],\n * starting from the specified [startIndex] toward the beginning.\n *\n * Returns `-1` if the specified [string] does not occur in this string builder starting at the specified [startIndex].\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun lastIndexOf(string: String, startIndex: Int): Int {\n if (string.isEmpty() && startIndex < 0) return -1\n return this.string.asDynamic().lastIndexOf(string, startIndex)\n }\n\n /**\n * Inserts the string representation of the specified boolean [value] into this string builder at the specified [index] and returns this instance.\n *\n * The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method,\n * and then that string was inserted into this string builder at the specified [index].\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun insert(index: Int, value: Boolean): StringBuilder {\n AbstractList.checkPositionIndex(index, length)\n\n string = string.substring(0, index) + value + string.substring(index)\n return this\n }\n\n /**\n * Inserts the specified character [value] into this string builder at the specified [index] and returns this instance.\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun insert(index: Int, value: Char): StringBuilder {\n AbstractList.checkPositionIndex(index, length)\n\n string = string.substring(0, index) + value + string.substring(index)\n return this\n }\n\n /**\n * Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance.\n *\n * The inserted characters go in same order as in the [value] character array, starting at [index].\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun insert(index: Int, value: CharArray): StringBuilder {\n AbstractList.checkPositionIndex(index, length)\n\n string = string.substring(0, index) + value.concatToString() + string.substring(index)\n return this\n }\n\n /**\n * Inserts characters in the specified character sequence [value] into this string builder at the specified [index] and returns this instance.\n *\n * The inserted characters go in the same order as in the [value] character sequence, starting at [index].\n *\n * @param index the position in this string builder to insert at.\n * @param value the character sequence from which characters are inserted. If [value] is `null`, then the four characters `\"null\"` are inserted.\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun insert(index: Int, value: CharSequence?): StringBuilder {\n AbstractList.checkPositionIndex(index, length)\n\n string = string.substring(0, index) + value.toString() + string.substring(index)\n return this\n }\n\n /**\n * Inserts the string representation of the specified object [value] into this string builder at the specified [index] and returns this instance.\n *\n * The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method,\n * and then that string was inserted into this string builder at the specified [index].\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun insert(index: Int, value: Any?): StringBuilder {\n AbstractList.checkPositionIndex(index, length)\n\n string = string.substring(0, index) + value.toString() + string.substring(index)\n return this\n }\n\n @Deprecated(\"Provided for binary compatibility.\", level = DeprecationLevel.HIDDEN)\n fun insert(index: Int, value: String): StringBuilder = insert(index, value)\n\n /**\n * Inserts the string [value] into this string builder at the specified [index] and returns this instance.\n *\n * If [value] is `null`, then the four characters `\"null\"` are inserted.\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n actual fun insert(index: Int, value: String?): StringBuilder {\n AbstractList.checkPositionIndex(index, length)\n\n val toInsert = value ?: \"null\"\n this.string =

```
this.string.substring(0, index) + toInsert + this.string.substring(index)\n        return this\n    }\n\n    /**\n     * Sets the length of this string builder to the specified [newLength].\n     *\n     * If the [newLength] is less than the current length, it is changed to the specified [newLength].\n     * Otherwise, null characters '\\u0000' are appended to this string builder until its length is less than the [newLength].\n     *\n     * Note that in Kotlin/JS [set] operator function has non-constant execution time complexity.\n     * Therefore, increasing length of this string builder and then updating each character by index may slow down your program.\n     *\n     * @throws IndexOutOfBoundsException or [IllegalArgumentException] if [newLength] is less than zero.\n     */\n    @SinceKotlin(\"1.4\")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun setLength(newLength: Int) {\n        if (newLength < 0) {\n            throw IllegalArgumentException(\"Negative new length: $newLength.\")\n        }\n\n        if (newLength <= length) {\n            string = string.substring(0, newLength)\n        } else {\n            for (i in length until newLength) {\n                string += '\\u0000'\n            }\n        }\n    }\n\n    /**\n     * Returns a new [String] that contains characters in this string builder at [startIndex] (inclusive) and up to the [length] (exclusive).\n     *\n     * @throws IndexOutOfBoundsException if [startIndex] is less than zero or greater than the length of this string builder.\n     */\n    @SinceKotlin(\"1.4\")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun substring(startIndex: Int): String {\n        AbstractList.checkPositionIndex(startIndex, length)\n\n        return string.substring(startIndex)\n    }\n\n    /**\n     * Returns a new [String] that contains characters in this string builder at [startIndex] (inclusive) and up to the [endIndex] (exclusive).\n     *\n     * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n     */\n    @SinceKotlin(\"1.4\")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun substring(startIndex: Int, endIndex: Int): String {\n        AbstractList.checkBoundsIndexes(startIndex, endIndex, length)\n\n        return string.substring(startIndex, endIndex)\n    }\n\n    /**\n     * Attempts to reduce storage used for this string builder.\n     *\n     * If the backing storage of this string builder is larger than necessary to hold its current contents,\n     * then it may be resized to become more space efficient.\n     * Calling this method may, but is not required to, affect the value of the [capacity] property.\n     *\n     * In Kotlin/JS implementation of StringBuilder the size of the backing storage is always equal to the length of the string builder.\n     */\n    @SinceKotlin(\"1.4\")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun trimToSize() {\n    }\n\n    override fun toString(): String = string\n\n    /**\n     * Clears the content of this string builder making it empty and returns this instance.\n     *\n     * @sample samples.text.Strings.clearStringBuilder\n     */\n    @SinceKotlin(\"1.3\")\n    public fun clear(): StringBuilder {\n        string = \"\"\n        return this\n    }\n\n    /**\n     * Sets the character at the specified [index] to the specified [value].\n     *\n     * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n     */\n    @SinceKotlin(\"1.4\")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    public operator fun set(index: Int, value: Char) {\n        AbstractList.checkElementIndex(index, length)\n\n        string = string.substring(0, index) + value + string.substring(index + 1)\n    }\n\n    /**\n     * Replaces characters in the specified range of this string builder with characters in the specified string [value] and returns this instance.\n     *\n     * @param startIndex the beginning (inclusive) of the range to replace.\n     * @param endIndex the end (exclusive) of the range to replace.\n     * @param value the string to replace with.\n     *\n     * @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] is less than zero, greater than the length of this string builder, or `startIndex > endIndex`.\n     */\n    @SinceKotlin(\"1.4\")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    public fun setRange(startIndex: Int, endIndex: Int, value: String): StringBuilder {\n        checkReplaceRange(startIndex, endIndex, length)\n\n        this.string = this.string.substring(0, startIndex) + value + this.string.substring(endIndex)\n        return this\n    }\n\n    private fun checkReplaceRange(startIndex: Int, endIndex: Int, length: Int) {\n        if (startIndex < 0 || startIndex > length) {\n            throw IndexOutOfBoundsException(\"startIndex: $startIndex, length: $length\")\n        }\n        if (startIndex > endIndex) {\n            throw IllegalArgumentException(\"startIndex($startIndex) > endIndex($endIndex)\")\n        }\n    }\n\n    /**\n     * Removes the character at the specified [index] from this string builder and returns this instance.\n     *\n     * If the `Char` at the specified [index] is part of a supplementary code point, this method does not remove the entire supplementary character.\n     *\n     * @param index the index of `Char` to remove.\n     *\n     * @throws
```

IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n   */\n   @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n   public fun deleteAt(index: Int): StringBuilder {\n AbstractList.checkElementIndex(index, length)\n\n   string = string.substring(0, index) + string.substring(index + 1)\n   return this\n   }\n\n   /**\n   * Removes characters in the specified range from this string builder and returns this instance.\n   *\n   * @param startIndex the beginning (inclusive) of the range to remove.\n   * @param endIndex the end (exclusive) of the range to remove.\n   *\n   * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n   */\n   @SinceKotlin(\"1.4\")\n   @WasExperimental(ExperimentalStdlibApi::class)\n   public fun deleteRange(startIndex: Int, endIndex: Int): StringBuilder {\n   checkReplaceRange(startIndex, endIndex, length)\n\n   string = string.substring(0, startIndex) + string.substring(endIndex)\n   return this\n   }\n\n   /**\n   * Copies characters from this string builder into the [destination] character array.\n   *\n   * @param destination the array to copy to.\n   * @param destinationOffset the position in the array to copy to, 0 by default.\n   * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.\n   * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.\n   *\n   * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n   * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n   * or when that index is out of the [destination] array indices range.\n   */\n   @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n   public fun toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) {\n AbstractList.checkBoundsIndexes(startIndex, endIndex, length)\n AbstractList.checkBoundsIndexes(destinationOffset, destinationOffset + endIndex - startIndex, destination.size)\n\n   var dstIndex = destinationOffset\n   for (index in startIndex until endIndex) {\n   destination[dstIndex++] = string[index]\n   }\n   }\n\n   /**\n   * Appends characters in a subarray of the specified character array [value] to this string builder and returns this instance.\n   *\n   * Characters are appended in order, starting at specified [startIndex].\n   *\n   * @param value the array from which characters are appended.\n   * @param startIndex the beginning (inclusive) of the subarray to append.\n   * @param endIndex the end (exclusive) of the subarray to append.\n   *\n   * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.\n   */\n   @SinceKotlin(\"1.4\")\n   @WasExperimental(ExperimentalStdlibApi::class)\n   public fun appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder {\n   string += value.concatToString(startIndex, endIndex)\n   return this\n   }\n\n   /**\n   * Appends a subsequence of the specified character sequence [value] to this string builder and returns this instance.\n   *\n   * @param value the character sequence from which a subsequence is appended.\n   * @param startIndex the beginning (inclusive) of the subsequence to append.\n   * @param endIndex the end (exclusive) of the subsequence to append.\n   *\n   * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.\n   */\n   @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n   public fun appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder {\n   val stringCsq = value.toString()\n AbstractList.checkBoundsIndexes(startIndex, endIndex, stringCsq.length)\n\n   string += stringCsq.substring(startIndex, endIndex)\n   return this\n   }\n\n   /**\n   * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.\n   *\n   * The inserted characters go in same order as in the [value] array, starting at [index].\n   *\n   * @param index the position in this string builder to insert at.\n   * @param value the array from which characters are inserted.\n * @param startIndex the beginning (inclusive) of the subarray to insert.\n   * @param endIndex the end (exclusive) of the subarray to insert.\n   *\n   * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.\n   * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n

```
*/\n    @SinceKotlin(\"1.4\")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    public fun insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder {\n AbstractList.checkPositionIndex(index, this.length)\n\n        string = string.substring(0, index) + value.concatToString(startIndex, endIndex) + string.substring(index)\n        return this\n    }\n\n    /**\n     * Inserts characters in a subsequence of the specified character sequence [value] into this string builder at the specified [index] and returns this instance.\n     *\n     * The inserted characters go in the same order as in the [value] character sequence, starting at [index].\n     *\n     * @param index the position in this string builder to insert at.\n     * @param value the character sequence from which a subsequence is inserted.\n     * @param startIndex the beginning (inclusive) of the subsequence to insert.\n     * @param endIndex the end (exclusive) of the subsequence to insert.\n     *\n     * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.\n     * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n     */\n    @SinceKotlin(\"1.4\")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    public fun insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder {\n        AbstractList.checkPositionIndex(index, length)\n\n        val stringCsq = value.toString()\n        AbstractList.checkBoundsIndexes(startIndex, endIndex, stringCsq.length)\n\n        string = string.substring(0, index) + stringCsq.substring(startIndex, endIndex) + string.substring(index)\n        return this\n    }\n}\n\n\n/**\n * Clears the content of this string builder making it empty and returns this instance.\n *\n * @sample samples.text.Strings.clearStringBuilder\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\", \"NOTHING_TO_INLINE\")\npublic actual inline fun StringBuilder.clear(): StringBuilder = this.clear()\n\n/**\n * Sets the character at the specified [index] to the specified [value].\n *\n * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\", \"NOTHING_TO_INLINE\")\npublic actual inline operator fun StringBuilder.set(index: Int, value: Char) = this.set(index, value)\n\n/**\n * Replaces characters in the specified range of this string builder with characters in the specified string [value] and returns this instance.\n *\n * @param startIndex the beginning (inclusive) of the range to replace.\n * @param endIndex the end (exclusive) of the range to replace.\n * @param value the string to replace with.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] is less than zero, greater than the length of this string builder, or `startIndex > endIndex`.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\", \"NOTHING_TO_INLINE\")\npublic actual inline fun StringBuilder.setRange(startIndex: Int, endIndex: Int, value: String): StringBuilder =\n    this.setRange(startIndex, endIndex, value)\n\n/**\n * Removes the character at the specified [index] from this string builder and returns this instance.\n *\n * If the `Char` at the specified [index] is part of a supplementary code point, this method does not remove the entire supplementary character.\n *\n * @param index the index of `Char` to remove.\n *\n * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\", \"NOTHING_TO_INLINE\")\npublic actual inline fun StringBuilder.deleteAt(index: Int): StringBuilder = this.deleteAt(index)\n\n/**\n * Removes characters in the specified range from this string builder and returns this instance.\n *\n * @param startIndex the beginning (inclusive) of the range to remove.\n * @param endIndex the end (exclusive) of the range to remove.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\", \"NOTHING_TO_INLINE\")\npublic actual inline fun StringBuilder.deleteRange(startIndex: Int, endIndex: Int): StringBuilder = this.deleteRange(startIndex, endIndex)\n\n/**\n * Copies characters from this string builder into the [destination] character array.\n *\n *
```

@param destination the array to copy to.\n * @param destinationOffset the position in the array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.\n * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\", \"NOTHING_TO_INLINE\", \"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual inline fun StringBuilder.toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) =\n   this.toCharArray(destination, destinationOffset, startIndex, endIndex)\n\n/**\n * Appends characters in a subarray of the specified character array [value] to this string builder and returns this instance.\n *\n * Characters are appended in order, starting at specified [startIndex].\n *\n * @param value the array from which characters are appended.\n * @param startIndex the beginning (inclusive) of the subarray to append.\n * @param endIndex the end (exclusive) of the subarray to append.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\", \"NOTHING_TO_INLINE\")\npublic actual inline fun StringBuilder.appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder =\n   this.appendRange(value, startIndex, endIndex)\n\n/**\n * Appends a subsequence of the specified character sequence [value] to this string builder and returns this instance.\n *\n * @param value the character sequence from which a subsequence is appended.\n * @param startIndex the beginning (inclusive) of the subsequence to append.\n * @param endIndex the end (exclusive) of the subsequence to append.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\", \"NOTHING_TO_INLINE\")\npublic actual inline fun StringBuilder.appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder =\n   this.appendRange(value, startIndex, endIndex)\n\n/**\n * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.\n *\n * The inserted characters go in same order as in the [value] array, starting at [index].\n *\n * @param index the position in this string builder to insert at.\n * @param value the array from which characters are inserted.\n * @param startIndex the beginning (inclusive) of the subarray to insert.\n * @param endIndex the end (exclusive) of the subarray to insert.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\", \"NOTHING_TO_INLINE\")\npublic actual inline fun StringBuilder.insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder =\n   this.insertRange(index, value, startIndex, endIndex)\n\n/**\n * Inserts characters in a subsequence of the specified character sequence [value] into this string builder at the specified [index] and returns this instance.\n *\n * The inserted characters go in the same order as in the [value] character sequence, starting at [index].\n *\n * @param index the position in this string builder to insert at.\n * @param value the character sequence from which a subsequence is inserted.\n * @param startIndex the beginning (inclusive) of the subsequence to insert.\n * @param endIndex the end (exclusive) of the subsequence to insert.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence

indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\", \"NOTHING_TO_INLINE\")\npublic actual inline fun StringBuilder.insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder =\n this.insertRange(index, value, startIndex, endIndex)\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n\n/**\n * Returns `true` if the content of this string is equal to the word \"true\", ignoring case, and `false` otherwise.\n */\n@Deprecated(\"Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@kotlin.internal.InlineOnly\npublic actual inline fun String.toBoolean(): Boolean = this.toBoolean()\n\n/**\n * Returns `true` if this string is not `null` and its content is equal to the word \"true\", ignoring case, and `false` otherwise.\n *\n * There are also strict versions of the function available on non-nullable String, [toBooleanStrict] and [toBooleanStrictOrNull].\n */\n@SinceKotlin(\"1.4\")\npublic actual fun String?.toBoolean(): Boolean = this != null && this.lowercase() == \"true\"\n\n/**\n * Parses the string as a signed [Byte] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */\npublic actual fun String.toByte(): Byte = toByteOrNull() ?: numberFormatError(this)\n\n/**\n * Parses the string as a signed [Byte] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\npublic actual fun String.toByte(radix: Int): Byte = toByteOrNull(radix) ?: numberFormatError(this)\n\n\n/**\n * Parses the string as a [Short] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */\npublic actual fun String.toShort(): Short = toShortOrNull() ?: numberFormatError(this)\n\n/**\n * Parses the string as a [Short] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\npublic actual fun String.toShort(radix: Int): Short = toShortOrNull(radix) ?: numberFormatError(this)\n\n/**\n * Parses the string as an [Int] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */\npublic actual fun String.toInt(): Int = toIntOrNull() ?: numberFormatError(this)\n\n/**\n * Parses the string as an [Int] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\npublic actual fun String.toInt(radix: Int): Int = toIntOrNull(radix) ?: numberFormatError(this)\n\n/**\n * Parses the string as a [Long] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */\npublic actual fun String.toLong(): Long = toLongOrNull() ?: numberFormatError(this)\n\n/**\n * Parses the string as a [Long] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\npublic actual fun String.toLong(radix: Int): Long = toLongOrNull(radix) ?: numberFormatError(this)\n\n/**\n * Parses the string as a [Double] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */\npublic actual fun String.toDouble(): Double = (+(this.asDynamic())).unsafeCast<Double>().also {\n   if (it.isNaN() && !this.isNaN() || it == 0.0 && this.isBlank())\n   numberFormatError(this)\n}\n\n/**\n * Parses the string as a [Float] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */\n@kotlin.internal.InlineOnly\npublic actual inline fun String.toFloat(): Float = toDouble().unsafeCast<Float>()\n\n/**\n * Parses the string as a [Double] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n */\npublic actual fun String.toDoubleOrNull(): Double? = (+(this.asDynamic())).unsafeCast<Double>().takeIf {\n   !(it.isNaN() && !this.isNaN() || it == 0.0 && this.isBlank())\n}\n\n/**\n * Parses the string as a [Float] number and returns the result\n * or `null` if the string is

not a valid representation of a number.\n */\n@kotlin.internal.InlineOnly\npublic actual inline fun String.toFloatOrNull(): Float? = toDoubleOrNull().unsafeCast<Float?>()\n\n/**\n * Returns a string representation of this [Byte] value in the specified [radix].\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n */\n@SinceKotlin(\"1.2\")\n@kotlin.internal.InlineOnly\npublic actual inline fun Byte.toString(radix: Int): String = this.toInt().toString(radix)\n\n/**\n * Returns a string representation of this [Short] value in the specified [radix].\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n */\n@SinceKotlin(\"1.2\")\n@kotlin.internal.InlineOnly\npublic actual inline fun Short.toString(radix: Int): String = this.toInt().toString(radix)\n\n/**\n * Returns a string representation of this [Int] value in the specified [radix].\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n */\n@SinceKotlin(\"1.2\")\npublic actual fun Int.toString(radix: Int): String = asDynamic().toString(checkRadix(radix))\n\nprivate fun String.isNaN(): Boolean = when (this.lowercase()) {\n \"nan\", \"+nan\", \"-nan\" -> true\n else -> false\n}\n\n/**\n * Checks whether the given [radix] is valid radix for string to number and number to string conversion.\n */\n@PublishedApi\ninternal actual fun checkRadix(radix: Int): Int {\n if (radix !in 2..36) {\n throw IllegalArgumentException(\"radix $radix was not in valid range 2..36\")\n }\n return radix\n}\n\ninternal actual fun digitOf(char: Char, radix: Int): Int = when {\n char >= '0' && char <= '9' -> char - '0'\n char >= 'A' && char <= 'Z' -> char - 'A' + 10\n char >= 'a' && char <= 'z' -> char - 'a' + 10\n char < '\\u0080' -> -1\n char >= '\\uFF21' && char <= '\\uFF3A' -> char - '\\uFF21' + 10 // full-width latin capital letter\n char >= '\\uFF41' && char <= '\\uFF5A' -> char - '\\uFF41' + 10 // full-width latin small letter\n else -> char.digitToIntImpl()\n}.let { if (it >= radix) -1 else it }\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\nimport kotlin.js.RegExp\n\n/**\n * Provides enumeration values to use to set regular expression options.\n */\npublic actual enum class RegexOption(val value: String) {\n /** Enables case-insensitive matching. */\n IGNORE_CASE(\"i\"),\n /** Enables multiline mode.\n *\n * In multiline mode the expressions `^` and `$` match just after or just before,\n * respectively, a line terminator or the end of the input sequence. */\n MULTILINE(\"m\")\n}\n\nprivate fun Iterable<RegexOption>.toFlags(prepend: String): String = joinToString(\"\", prefix = prepend) { it.value }\n\n\n/**\n * Represents the results from a single capturing group within a [MatchResult] of [Regex].\n *\n * @param value The value of captured group.\n */\npublic actual data class MatchGroup(actual val value: String)\n\n\n/**\n * Represents a compiled regular expression.\n * Provides functions to match strings in text with a pattern, replace the found occurrences and split text around matches.\n *\n * For pattern syntax reference see [MDN RegExp](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/RegExp#Special_characters_meaning_in_regular_expressions)\n * and [http://www.w3schools.com/jsref/jsref_obj_regexp.asp](https://www.w3schools.com/jsref/jsref_obj_regexp.asp).\n *\n * Note that `RegExp` objects under the hood are constructed with [the \"u\" flag](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/RegExp/unicode)\n * that enables Unicode-related features in regular expressions. This also makes the pattern syntax more strict,\n * for example, prohibiting unnecessary escape sequences.\n *\n * @constructor Creates a regular expression from the specified [pattern] string and the specified set of [options].\n */\npublic actual class Regex actual constructor(pattern: String, options: Set<RegexOption>) {\n\n /** Creates a regular expression from the specified [pattern] string and the specified single [option]. */\n public actual constructor(pattern: String, option: RegexOption) : this(pattern, setOf(option))\n\n /** Creates a regular expression from the specified [pattern] string and the default options. */\n public actual constructor(pattern: String) : this(pattern, emptySet())\n\n /** The pattern string of this regular expression. */\n public actual val pattern: String = pattern\n /** The set of options that were used to create this regular expression. */\n public actual val options: Set<RegexOption> = options.toSet()\n private val nativePattern: RegExp = RegExp(pattern, options.toFlags(\"gu\"))\n private var nativeStickyPattern: RegExp? = null\n private fun initStickyPattern(): RegExp =\n nativeStickyPattern ?: RegExp(pattern, options.toFlags(\"yu\")).also { nativeStickyPattern = it }\n\n private var

```
nativeMatchesEntirePattern: RegExp? = null\n    private fun initMatchesEntirePattern(): RegExp =\n nativeMatchesEntirePattern ?: run {\n            if (pattern.startsWith('^') && pattern.endsWith('$'))\n nativePattern\n            else\n                return RegExp(\"^${pattern.trimStart('^').trimEnd('$')}$\", options.toFlags(\"gu\"))\n        }.also { nativeMatchesEntirePattern = it }\n\n\n    /** Indicates whether the regular expression matches the entire [input]. */\n    public actual infix fun matches(input: CharSequence): Boolean {\n nativePattern.reset()\n        val match = nativePattern.exec(input.toString())\n        return match != null && match.index == 0 && nativePattern.lastIndex == input.length\n    }\n\n    /** Indicates whether the regular expression can find at least one match in the specified [input]. */\n    public actual fun containsMatchIn(input: CharSequence): Boolean {\n        nativePattern.reset()\n        return nativePattern.test(input.toString())\n    }\n\n    @SinceKotlin(\"1.5\")\n    @ExperimentalStdlibApi\n    public actual fun matchesAt(input: CharSequence, index: Int): Boolean {\n        if (index < 0 || index > input.length) {\n            throw IndexOutOfBoundsException(\"index out of bounds: $index, input length: ${input.length}\")\n        }\n        val pattern = initStickyPattern()\n pattern.lastIndex = index\n        return pattern.test(input.toString())\n    }\n\n    /**\n     * Returns the first match of a regular expression in the [input], beginning at the specified [startIndex].\n     *\n     * @param startIndex An index to start search with, by default 0. Must be not less than zero and not greater than `input.length()`\n     * @return An instance of [MatchResult] if match was found or `null` otherwise.\n     * @throws IndexOutOfBoundsException if [startIndex] is less than zero or greater than the length of the [input] char sequence.\n     * @sample samples.text.Regexps.find\n     */\n    @Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\n    public actual fun find(input: CharSequence, startIndex: Int = 0): MatchResult? {\n        if (startIndex < 0 || startIndex > input.length) {\n            throw IndexOutOfBoundsException(\"Start index out of bounds: $startIndex, input length: ${input.length}\")\n        }\n        return nativePattern.findNext(input.toString(), startIndex, nativePattern)\n    }\n\n    /**\n     * Returns a sequence of all occurrences of a regular expression within the [input] string, beginning at the specified [startIndex].\n     *\n     * @throws IndexOutOfBoundsException if [startIndex] is less than zero or greater than the length of the [input] char sequence.\n     *\n     * @sample samples.text.Regexps.findAll\n     */\n    @Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\n    public actual fun findAll(input: CharSequence, startIndex: Int = 0): Sequence<MatchResult> {\n        if (startIndex < 0 || startIndex > input.length) {\n            throw IndexOutOfBoundsException(\"Start index out of bounds: $startIndex, input length: ${input.length}\")\n        }\n        return generateSequence({ find(input, startIndex) }, { match -> match.next() })\n    }\n\n    /**\n     * Attempts to match the entire [input] CharSequence against the pattern.\n     *\n     * @return An instance of [MatchResult] if the entire input matches or `null` otherwise.\n     */\n    public actual fun matchEntire(input: CharSequence): MatchResult? =\n        initMatchesEntirePattern().findNext(input.toString(), 0, nativePattern)\n\n    @SinceKotlin(\"1.5\")\n    @ExperimentalStdlibApi\n    public actual fun matchAt(input: CharSequence, index: Int): MatchResult? {\n        if (index < 0 || index > input.length) {\n            throw IndexOutOfBoundsException(\"index out of bounds: $index, input length: ${input.length}\")\n        }\n        return initStickyPattern().findNext(input.toString(), index, nativePattern)\n    }\n\n\n    /**\n     * Replaces all occurrences of this regular expression in the specified [input] string with specified [replacement] expression.\n     *\n     * The replacement string may contain references to the captured groups during a match. Occurrences of `$index`\n     * in the replacement string will be substituted with the subsequences corresponding to the captured groups with the specified index.\n     * The first digit after '$' is always treated as part of group reference. Subsequent digits are incorporated\n     * into `index` only if they would form a valid group reference. Only the digits '0'..'9' are considered as potential components\n     * of the group reference. Note that indexes of captured groups start from 1, and the group with index 0 is the whole match.\n     *\n     * Backslash character '\\' can be used to include the succeeding character as a literal in the replacement string, e.g, `\\$` or `\\\\`.\n     * [Regex.escapeReplacement] can be used if [replacement] have to be treated as a literal string.\n     *\n     * Note that referring named capturing groups by name is currently not supported in Kotlin/JS.\n     * However, you can still refer them by index.\n     *\n     * @param input the char sequence to find matches of this regular expression in\n     * @param replacement the expression to replace found matches with\n     * @return the result of replacing each occurrence of this regular expression in [input] with
```

```
the result of evaluating the [replacement] expression\n     * @throws RuntimeException if [replacement] expression is malformed, or capturing group with specified `name` or `index` does not exist\n     */\n    public actual fun replace(input: CharSequence, replacement: String): String {\n        if (!replacement.contains('\\\\') && !replacement.contains('$')) {\n            return input.toString().nativeReplace(nativePattern, replacement)\n        }\n        return replace(input) { substituteGroupRefs(it, replacement) }\n    }\n\n    /**\n     * Replaces all occurrences of this regular expression in the specified [input] string with the result of\n     * the given function [transform] that takes [MatchResult] and returns a string to be used as a\n     * replacement for that match.\n     */\n    public actual fun replace(input: CharSequence, transform: (MatchResult) -> CharSequence): String {\n        var match = find(input)\n        if (match == null) return input.toString()\n\n        var lastStart = 0\n        val length = input.length\n        val sb = StringBuilder(length)\n        do {\n            val foundMatch = match!!\n            sb.append(input, lastStart, foundMatch.range.start)\n            sb.append(transform(foundMatch))\n            lastStart = foundMatch.range.endInclusive + 1\n            match = foundMatch.next()\n        } while (lastStart < length && match != null)\n\n        if (lastStart < length) {\n            sb.append(input, lastStart, length)\n        }\n\n        return sb.toString()\n    }\n\n    /**\n     * Replaces the first occurrence of this regular expression in the specified [input] string with specified [replacement] expression.\n     *\n     * The replacement string may contain references to the captured groups during a match. Occurrences of `$index`\n     * in the replacement string will be substituted with the subsequences corresponding to the captured groups with the specified index.\n     * The first digit after '$' is always treated as part of group reference. Subsequent digits are incorporated\n     * into `index` only if they would form a valid group reference. Only the digits '0'..'9' are considered as potential components\n     * of the group reference. Note that indexes of captured groups start from 1, and the group with index 0 is the whole match.\n     *\n     * Backslash character '\\' can be used to include the succeeding character as a literal in the replacement string, e.g, `\\$` or `\\\\`.\n     * [Regex.escapeReplacement] can be used if [replacement] have to be treated as a literal string.\n     *\n     * Note that referring named capturing groups by name is not supported currently in Kotlin/JS.\n     * However, you can still refer them by index.\n     *\n     * @param input the char sequence to find a match of this regular expression in\n     * @param replacement the expression to replace the found match with\n     * @return the result of replacing the first occurrence of this regular expression in [input] with the result of evaluating the [replacement] expression\n     * @throws RuntimeException if [replacement] expression is malformed, or capturing group with specified `name` or `index` does not exist\n     */\n    public actual fun replaceFirst(input: CharSequence, replacement: String): String {\n        if (!replacement.contains('\\\\') && !replacement.contains('$')) {\n            val nonGlobalOptions = options.toFlags(\"u\")\n            return input.toString().nativeReplace(RegExp(pattern, nonGlobalOptions), replacement)\n        }\n\n        val match = find(input) ?: return input.toString()\n\n        return buildString {\n            append(input.substring(0, match.range.first))\n            append(substituteGroupRefs(match, replacement))\n            append(input.substring(match.range.last + 1, input.length))\n        }\n    }\n\n    /**\n     * Splits the [input] CharSequence to a list of strings around matches of this regular expression.\n     *\n     * @param limit Non-negative value specifying the maximum number of substrings the string can be split to.\n     * Zero by default means no limit is set.\n     */\n    @Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\n    public actual fun split(input: CharSequence, limit: Int = 0): List<String> {\n        requireNonNegativeLimit(limit)\n        val matches = findAll(input).let { if (limit == 0) it else it.take(limit - 1) }\n        val result = mutableListOf<String>()\n        var lastStart = 0\n\n        for (match in matches) {\n            result.add(input.subSequence(lastStart, match.range.start).toString())\n            lastStart = match.range.endInclusive + 1\n        }\n        result.add(input.subSequence(lastStart, input.length).toString())\n        return result\n    }\n\n    /**\n     * Splits the [input] CharSequence to a sequence of strings around matches of this regular expression.\n     *\n     * @param limit Non-negative value specifying the maximum number of substrings the string can be split to.\n     * Zero by default means no limit is set.\n     * @sample samples.text.Regexps.splitToSequence\n     */\n    @SinceKotlin(\"1.6\")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    @Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\n    public actual fun splitToSequence(input: CharSequence, limit: Int = 0): Sequence<String> {\n        requireNonNegativeLimit(limit)\n\n        return sequence {\n            var match = find(input)\n            if (match ==
```

```
null || limit == 1) {\n yield(input.toString())\n return@sequence\n }\n\n var
nextStart = 0\n var splitCount = 0\n\n do {\n val foundMatch = match!!\n
yield(input.substring(nextStart, foundMatch.range.first))\n nextStart = foundMatch.range.endInclusive +
1\n match = foundMatch.next()\n } while (++splitCount != limit - 1 && match != null)\n\n
yield(input.substring(nextStart, input.length))\n }\n }\n\n\n /**\n * Returns the string representation of
this regular expression, namely the [pattern] of this regular expression.\n *\n * Note that another regular
expression constructed from the same pattern string may have different [options]\n * and may match strings
differently.\n */\n public override fun toString(): String = nativePattern.toString()\n\n actual companion object
{\n /**\n * Returns a regular expression that matches the specified [literal] string literally.\n * No
characters of that string will have special meaning when searching for an occurrence of the regular expression.\n
*/\n public actual fun fromLiteral(literal: String): Regex = Regex(escape(literal))\n\n /**\n * Returns a
regular expression pattern string that matches the specified [literal] string literally.\n * No characters of that
string will have special meaning when searching for an occurrence of the regular expression.\n */\n public
actual fun escape(literal: String): String = literal.nativeReplace(patternEscape, \"\\\\$&\")\n\n /**\n *
Returns a literal replacement expression for the specified [literal] string.\n * No characters of that string will
have special meaning when it is used as a replacement string in [Regex.replace] function.\n */\n public
actual fun escapeReplacement(literal: String): String = literal.nativeReplace(replacementEscape, \"\\\\$&\")\n\n
private val patternEscape = RegExp(\"\"\"[\\\\^$*+?.()|[\\]{}]\"\"\", \"g\")\n private val replacementEscape =
RegExp(\"\"\"[\\\\$]\"\"\", \"g\")\n\n internal fun nativeEscapeReplacement(literal: String): String =
literal.nativeReplace(nativeReplacementEscape, \"$$$$\")\n private val nativeReplacementEscape =
RegExp(\"\"\"\\$\"\"\", \"g\")\n }\n}\n\n\n\nprivate fun RegExp.findNext(input: String, from: Int, nextPattern:
RegExp): MatchResult? {\n this.lastIndex = from\n val match = exec(input)\n if (match == null) return null\n
val range = match.index..lastIndex - 1\n\n return object : MatchResult {\n override val range: IntRange =
range\n override val value: String\n get() = match[0]!!\n\n override val groups:
MatchGroupCollection = object : MatchGroupCollection, AbstractCollection<MatchGroup?>() {\n override
val size: Int get() = match.length\n override fun iterator(): Iterator<MatchGroup?> =
indices.asSequence().map { this[it] }.iterator()\n override fun get(index: Int): MatchGroup? =
match[index]?.let { MatchGroup(it) }\n }\n\n\n private var groupValues_: List<String>? = null\n\n
override val groupValues: List<String>\n get() {\n if (groupValues_ == null) {\n
groupValues_ = object : AbstractList<String>() {\n override val size: Int get() = match.length\n
override fun get(index: Int): String = match[index] ?: \"\"\n }\n }\n return
groupValues_!!\n }\n\n override fun next(): MatchResult? =\n nextPattern.findNext(input, if
(range.isEmpty()) advanceToNextCharacter(range.start) else range.endInclusive + 1, nextPattern)\n\n private
fun advanceToNextCharacter(index: Int): Int {\n if (index < input.lastIndex) {\n val code1 =
input.asDynamic().charCodeAt(index).unsafeCast<Int>()\n if (code1 in 0xD800..0xDBFF) {\n
val code2 = input.asDynamic().charCodeAt(index + 1).unsafeCast<Int>()\n if (code2 in
0xDC00..0xDFFF) {\n return index + 2\n }\n }\n }\n return index +
1\n }\n }\n}\n\n// The same code from K/N Regex.kt\nprivate fun substituteGroupRefs(match: MatchResult,
replacement: String): String {\n var index = 0\n val result = StringBuilder(replacement.length)\n\n while
(index < replacement.length) {\n val char = replacement[index++]\n if (char == '\\') {\n if (index ==
replacement.length)\n throw IllegalArgumentException(\"The Char to be escaped is missing\")\n\n
result.append(replacement[index++])\n } else if (char == '$') {\n if (index == replacement.length)\n
throw IllegalArgumentException(\"Capturing group index is missing\")\n\n if (replacement[index] == '{')\n
throw IllegalArgumentException(\"Named capturing group reference currently is not supported\")\n\n
if (replacement[index] !in '0'..'9')\n throw IllegalArgumentException(\"Invalid capturing group
reference\")\n\n val endIndex = replacement.readGroupIndex(index, match.groupValues.size)\n val
groupIndex = replacement.substring(index, endIndex).toInt()\n\n if (groupIndex >=
match.groupValues.size)\n throw IndexOutOfBoundsException(\"Group with index $groupIndex does not
```

exist\")\n\n result.append(match.groupValues[groupIndex])\n index = endIndex\n } else {\n result.append(char)\n }\n }\n return result.toString()\n}\n\nprivate fun String.readGroupIndex(startIndex: Int, groupCount: Int): Int {\n // at least one digit after '$' is always captured\n var index = startIndex + 1\n var groupIndex = this[startIndex] - '0'\n\n // capture the largest valid group index\n while (index < length && this[index] in '0'..'9') {\n val newGroupIndex = (groupIndex * 10) + (this[index] - '0')\n if (newGroupIndex in 0 until groupCount) {\n groupIndex = newGroupIndex\n index++\n } else {\n break\n }\n }\n return index\n}","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"StringsKt\")\n@file:Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\")\n\npackage kotlin.text\n\nimport kotlin.contracts.*\n\n/**\n * A mutable sequence of characters.\n *\n * String builder can be used to efficiently perform multiple string manipulation operations.\n */\nexpect class StringBuilder : Appendable, CharSequence {\n /** Constructs an empty string builder. */\n constructor()\n\n /** Constructs an empty string builder with the specified initial [capacity]. */\n constructor(capacity: Int)\n\n /** Constructs a string builder that contains the same characters as the specified [content] char sequence. */\n constructor(content: CharSequence)\n\n /** Constructs a string builder that contains the same characters as the specified [content] string. */\n @SinceKotlin(\"1.3\")\n// @ExperimentalStdlibApi\n constructor(content: String)\n\n override val length: Int\n\n override operator fun get(index: Int): Char\n\n override fun subSequence(startIndex: Int, endIndex: Int): CharSequence\n\n override fun append(value: Char): StringBuilder\n override fun append(value: CharSequence?): StringBuilder\n override fun append(value: CharSequence?, startIndex: Int, endIndex: Int): StringBuilder\n\n /**\n * Reverses the contents of this string builder and returns this instance.\n *\n * Surrogate pairs included in this string builder are treated as single characters.\n * Therefore, the order of the high-low surrogates is never reversed.\n *\n * Note that the reverse operation may produce new surrogate pairs that were unpaired low-surrogates and high-surrogates before the operation.\n * For example, reversing `\"\\uDC00\\uD800\"` produces `\"\\uD800\\uDC00\"` which is a valid surrogate pair.\n */\n fun reverse(): StringBuilder\n\n /**\n * Appends the string representation of the specified object [value] to this string builder and returns this instance.\n *\n * The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method,\n * and then that string was appended to this string builder.\n */\n fun append(value: Any?): StringBuilder\n\n /**\n * Appends the string representation of the specified boolean [value] to this string builder and returns this instance.\n *\n * The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method,\n * and then that string was appended to this string builder.\n */\n @SinceKotlin(\"1.3\")\n fun append(value: Boolean): StringBuilder\n\n /**\n * Appends characters in the specified character array [value] to this string builder and returns this instance.\n *\n * Characters are appended in order, starting at the index 0.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun append(value: CharArray): StringBuilder\n\n /**\n * Appends the specified string [value] to this string builder and returns this instance.\n *\n * If [value] is `null`, then the four characters `\"null\"` are appended.\n */\n @SinceKotlin(\"1.3\")\n fun append(value: String?): StringBuilder\n\n /**\n * Returns the current capacity of this string builder.\n *\n * The capacity is the maximum length this string builder can have before an allocation occurs.\n */\n @SinceKotlin(\"1.3\")\n// @ExperimentalStdlibApi\n @Deprecated(\"Obtaining StringBuilder capacity is not supported in JS and common code.\", level = DeprecationLevel.ERROR)\n fun capacity(): Int\n\n /**\n * Ensures that the capacity of this string builder is at least equal to the specified [minimumCapacity].\n *\n * If the current capacity is less than the [minimumCapacity], a new backing storage is allocated with greater capacity.\n * Otherwise, this method takes no action and simply returns.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun ensureCapacity(minimumCapacity: Int)\n\n /**\n * Returns the index within this string builder of the first occurrence of the specified [string].\n *\n * Returns `-1` if the specified [string] does not occur in this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun indexOf(string: String): Int\n\n /**\n * Returns the

index within this string builder of the first occurrence of the specified [string],\n * starting at the specified [startIndex].\n *\n * Returns `-1` if the specified [string] does not occur in this string builder starting at the specified [startIndex].\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun indexOf(string: String, startIndex: Int): Int\n\n /**\n * Returns the index within this string builder of the last occurrence of the specified [string].\n * The last occurrence of empty string `\"\"` is considered to be at the index equal to `this.length`.\n *\n * Returns `-1` if the specified [string] does not occur in this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun lastIndexOf(string: String): Int\n\n /**\n * Returns the index within this string builder of the last occurrence of the specified [string],\n * starting from the specified [startIndex] toward the beginning.\n *\n * Returns `-1` if the specified [string] does not occur in this string builder starting at the specified [startIndex].\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun lastIndexOf(string: String, startIndex: Int): Int\n\n /**\n * Inserts the string representation of the specified boolean [value] into this string builder at the specified [index] and returns this instance.\n *\n * The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method,\n * and then that string was inserted into this string builder at the specified [index].\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun insert(index: Int, value: Boolean): StringBuilder\n\n /**\n * Inserts the specified character [value] into this string builder at the specified [index] and returns this instance.\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun insert(index: Int, value: Char): StringBuilder\n\n /**\n * Inserts characters in the specified character array [value] into this string builder at the specified [index] and returns this instance.\n *\n * The inserted characters go in same order as in the [value] character array, starting at [index].\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun insert(index: Int, value: CharArray): StringBuilder\n\n /**\n * Inserts characters in the specified character sequence [value] into this string builder at the specified [index] and returns this instance.\n *\n * The inserted characters go in the same order as in the [value] character sequence, starting at [index].\n *\n * @param index the position in this string builder to insert at.\n * @param value the character sequence from which characters are inserted. If [value] is `null`, then the four characters `\"null\"` are inserted.\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun insert(index: Int, value: CharSequence?): StringBuilder\n\n /**\n * Inserts the string representation of the specified object [value] into this string builder at the specified [index] and returns this instance.\n *\n * The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method,\n * and then that string was inserted into this string builder at the specified [index].\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun insert(index: Int, value: Any?): StringBuilder\n\n /**\n * Inserts the string [value] into this string builder at the specified [index] and returns this instance.\n *\n * If [value] is `null`, then the four characters `\"null\"` are inserted.\n *\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun insert(index: Int, value: String?): StringBuilder\n\n /**\n * Sets the length of this string builder to the specified [newLength].\n *\n * If the [newLength] is less than the current length, it is changed to the specified [newLength].\n * Otherwise, null characters '\\u0000' are appended to this string builder until its length is less than the [newLength].\n *\n * Note that in Kotlin/JS [set] operator function has non-constant execution time complexity.\n * Therefore, increasing length of this string builder and then updating each character by index may slow down your program.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] if [newLength] is less than zero.\n */\n @SinceKotlin(\"1.4\")\n @WasExperimental(ExperimentalStdlibApi::class)\n fun setLength(newLength:

Int)\n\n   /**\n   * Returns a new [String] that contains characters in this string builder at [startIndex] (inclusive) and up to the [length] (exclusive).\n   *\n   * @throws IndexOutOfBoundsException if [startIndex] is less than zero or greater than the length of this string builder.\n   */\n   @SinceKotlin(\"1.4\")\n   @WasExperimental(ExperimentalStdlibApi::class)\n   fun substring(startIndex: Int): String\n\n   /**\n   * Returns a new [String] that contains characters in this string builder at [startIndex] (inclusive) and up to the [endIndex] (exclusive).\n   *\n   * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n   */\n   @SinceKotlin(\"1.4\")\n   @WasExperimental(ExperimentalStdlibApi::class)\n   fun substring(startIndex: Int, endIndex: Int): String\n\n   /**\n   * Attempts to reduce storage used for this string builder.\n   *\n   * If the backing storage of this string builder is larger than necessary to hold its current contents,\n   * then it may be resized to become more space efficient.\n   * Calling this method may, but is not required to, affect the value of the [capacity] property.\n   */\n   @SinceKotlin(\"1.4\")\n   @WasExperimental(ExperimentalStdlibApi::class)\n   fun trimToSize()\n}\n\n\n/**\n * Clears the content of this string builder making it empty and returns this instance.\n *\n * @sample samples.text.Strings.clearStringBuilder\n */\n@SinceKotlin(\"1.3\")\npublic expect fun StringBuilder.clear(): StringBuilder\n\n/**\n * Sets the character at the specified [index] to the specified [value].\n *\n * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect operator fun StringBuilder.set(index: Int, value: Char)\n\n/**\n * Replaces characters in the specified range of this string builder with characters in the specified string [value] and returns this instance.\n *\n * @param startIndex the beginning (inclusive) of the range to replace.\n * @param endIndex the end (exclusive) of the range to replace.\n * @param value the string to replace with.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] is less than zero, greater than the length of this string builder, or `startIndex > endIndex`.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun StringBuilder.setRange(startIndex: Int, endIndex: Int, value: String): StringBuilder\n\n/**\n * Removes the character at the specified [index] from this string builder and returns this instance.\n *\n * If the `Char` at the specified [index] is part of a supplementary code point, this method does not remove the entire supplementary character.\n *\n * @param index the index of `Char` to remove.\n *\n * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun StringBuilder.deleteAt(index: Int): StringBuilder\n\n/**\n * Removes characters in the specified range from this string builder and returns this instance.\n *\n * @param startIndex the beginning (inclusive) of the range to remove.\n * @param endIndex the end (exclusive) of the range to remove.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun StringBuilder.deleteRange(startIndex: Int, endIndex: Int): StringBuilder\n\n/**\n * Copies characters from this string builder into the [destination] character array.\n *\n * @param destination the array to copy to.\n * @param destinationOffset the position in the array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.\n * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun StringBuilder.toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length)\n\n/**\n * Appends characters in a subarray of the specified character array [value] to this string builder and returns this instance.\n *\n * Characters are appended in order, starting at specified [startIndex].\n *\n * @param value the array from which characters are appended.\n * @param startIndex the beginning (inclusive) of the

subarray to append.\n * @param endIndex the end (exclusive) of the subarray to append.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun StringBuilder.appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder\n\n/**\n * Appends a subsequence of the specified character sequence [value] to this string builder and returns this instance.\n *\n * @param value the character sequence from which a subsequence is appended.\n * @param startIndex the beginning (inclusive) of the subsequence to append.\n * @param endIndex the end (exclusive) of the subsequence to append.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun StringBuilder.appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder\n\n/**\n * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.\n *\n * The inserted characters go in same order as in the [value] array, starting at [index].\n *\n * @param index the position in this string builder to insert at.\n * @param value the array from which characters are inserted.\n * @param startIndex the beginning (inclusive) of the subarray to insert.\n * @param endIndex the end (exclusive) of the subarray to insert.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun StringBuilder.insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder\n\n/**\n * Inserts characters in a subsequence of the specified character sequence [value] into this string builder at the specified [index] and returns this instance.\n *\n * The inserted characters go in the same order as in the [value] character sequence, starting at [index].\n *\n * @param index the position in this string builder to insert at.\n * @param value the character sequence from which a subsequence is inserted.\n * @param startIndex the beginning (inclusive) of the subsequence to insert.\n * @param endIndex the end (exclusive) of the subsequence to insert.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun StringBuilder.insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder\n\n@Suppress(\"EXTENSION_SHADOWED_BY_MEMBER\")\n@Deprecated(\"Use append(value: Any?) instead\", ReplaceWith(\"append(value = obj)\"), DeprecationLevel.WARNING)\n@kotlin.internal.InlineOnly\npublic inline fun StringBuilder.append(obj: Any?): StringBuilder = this.append(obj)\n\n/**\n * Builds new string by populating newly created [StringBuilder] using provided [builderAction]\n * and then converting it to [String].\n */\n@kotlin.internal.InlineOnly\npublic inline fun buildString(builderAction: StringBuilder.() -> Unit): String {\n   contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }\n   return StringBuilder().apply(builderAction).toString()\n}\n\n/**\n * Builds new string by populating newly created [StringBuilder] initialized with the given [capacity]\n * using provided [builderAction] and then converting it to [String].\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun buildString(capacity: Int, builderAction: StringBuilder.() -> Unit): String {\n   contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }\n   return StringBuilder(capacity).apply(builderAction).toString()\n}\n\n/**\n * Appends all arguments to the given StringBuilder.\n */\npublic fun StringBuilder.append(vararg value: String?): StringBuilder {\n   for (item in value)\n   append(item)\n   return this\n}\n\n/**\n * Appends all arguments to the given StringBuilder.\n */\npublic fun StringBuilder.append(vararg value: Any?): StringBuilder {\n   for (item in value)\n   append(item)\n   return this\n}\n\n/** Appends a line feed character (`\\n`) to this StringBuilder.

```
*/\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun StringBuilder.appendLine():
StringBuilder = append('\\n')\n\n/** Appends [value] to this [StringBuilder], followed by a line feed character
(`\\n`). */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun StringBuilder.appendLine(value:
CharSequence?): StringBuilder = append(value).appendLine()\n\n/** Appends [value] to this [StringBuilder],
followed by a line feed character (`\\n`). */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun
StringBuilder.appendLine(value: String?): StringBuilder = append(value).appendLine()\n\n/** Appends [value] to
this [StringBuilder], followed by a line feed character (`\\n`).
*/\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun StringBuilder.appendLine(value: Any?):
StringBuilder = append(value).appendLine()\n\n/** Appends [value] to this [StringBuilder], followed by a line feed
character (`\\n`). */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun
StringBuilder.appendLine(value: CharArray): StringBuilder = append(value).appendLine()\n\n/** Appends [value]
to this [StringBuilder], followed by a line feed character (`\\n`).
*/\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun StringBuilder.appendLine(value: Char):
StringBuilder = append(value).appendLine()\n\n/** Appends [value] to this [StringBuilder], followed by a line feed
character (`\\n`). */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun
StringBuilder.appendLine(value: Boolean): StringBuilder = append(value).appendLine()\n","/*\n * Copyright 2010-
2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\nimport
kotlin.js.RegExp\n\n@kotlin.internal.InlineOnly\ninternal actual inline fun String.nativeIndexOf(ch: Char,
fromIndex: Int): Int = nativeIndexOf(ch.toString(), fromIndex)\n\n@kotlin.internal.InlineOnly\ninternal actual
inline fun String.nativeLastIndexOf(ch: Char, fromIndex: Int): Int = nativeLastIndexOf(ch.toString(),
fromIndex)\n\n/**\n * Returns `true` if this string starts with the specified prefix.\n
*/\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun
String.startsWith(prefix: String, ignoreCase: Boolean = false): Boolean {\n    if (!ignoreCase)\n        return
nativeStartsWith(prefix, 0)\n    else\n        return regionMatches(0, prefix, 0, prefix.length, ignoreCase)\n}\n\n/**\n *
Returns `true` if a substring of this string starting at the specified offset [startIndex] starts with the specified prefix.\n
*/\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun
String.startsWith(prefix: String, startIndex: Int, ignoreCase: Boolean = false): Boolean {\n    if (!ignoreCase)\n
return nativeStartsWith(prefix, startIndex)\n    else\n        return regionMatches(startIndex, prefix, 0, prefix.length,
ignoreCase)\n}\n\n/**\n * Returns `true` if this string ends with the specified suffix.\n
*/\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun
String.endsWith(suffix: String, ignoreCase: Boolean = false): Boolean {\n    if (!ignoreCase)\n        return
nativeEndsWith(suffix)\n    else\n        return regionMatches(length - suffix.length, suffix, 0, suffix.length,
ignoreCase)\n}\n\n@Deprecated(\"Use Regex.matches() instead\",
ReplaceWith(\"regex.toRegex().matches(this)\"))\n@DeprecatedSinceKotlin(warningSince = \"1.6\")\npublic fun
String.matches(regex: String): Boolean {\n    @Suppress(\"DEPRECATION\")\n    val result = this.match(regex)\n
return result != null && result.size != 0\n}\n\n/**\n * Returns `true` if this string is empty or consists solely of
whitespace characters.\n *\n * @sample samples.text.Strings.stringIsBlank\n */\npublic actual fun
CharSequence.isBlank(): Boolean = length == 0 || indices.all { this[it].isWhitespace() }\n\n/**\n * Returns `true` if
this string is equal to [other], optionally ignoring character case.\n *\n * Two strings are considered to be equal if
they have the same length and the same character at the same index.\n * If [ignoreCase] is true, the result of
`Char.uppercaseChar().lowercaseChar()` on each character is compared.\n *\n * @param ignoreCase `true` to ignore
character case when comparing strings. By default `false`.\n
*/\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun
String?.equals(other: String?, ignoreCase: Boolean = false): Boolean {\n    if (this == null) return other == null\n    if
(other == null) return false\n    if (!ignoreCase) return this == other\n\n    if (this.length != other.length) return
false\n\n    for (index in 0 until this.length) {\n        val thisChar = this[index]\n        val otherChar = other[index]\n
    if (!thisChar.equals(otherChar, ignoreCase)) {\n            return false\n        }\n    }\n\n    return
```

```
true\n}\n\n\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun CharSequence.regionMatches(thisOffset: Int, other: CharSequence, otherOffset: Int, length: Int, ignoreCase: Boolean = false): Boolean =\n    regionMatchesImpl(thisOffset, other, otherOffset, length, ignoreCase)\n\n\n/**\n * Returns a copy of this string having its first letter titlecased using the rules of the default locale,\n * or the original string if it's empty or already starts with a title case letter.\n *\n * The title case of a character is usually the same as its upper case with several exceptions.\n * The particular list of characters with the special title case form depends on the underlying platform.\n *\n * @sample samples.text.Strings.capitalize\n */\n@Deprecated(\"Use replaceFirstChar instead.\", ReplaceWith(\"replaceFirstChar { if (it.isLowerCase()) it.titlecase() else it.toString() }\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic actual fun String.capitalize(): String {\n    return if (isNotEmpty()) substring(0, 1).uppercase() + substring(1) else this\n}\n\n/**\n * Returns a copy of this string having its first letter lowercased using the rules of the default locale,\n * or the original string if it's empty or already starts with a lower case letter.\n *\n * @sample samples.text.Strings.decapitalize\n */\n@Deprecated(\"Use replaceFirstChar instead.\", ReplaceWith(\"replaceFirstChar { it.lowercase() }\"))\n@DeprecatedSinceKotlin(warningSince = \"1.5\")\npublic actual fun String.decapitalize(): String {\n    return if (isNotEmpty()) substring(0, 1).lowercase() + substring(1) else this\n}\n\n/**\n * Returns a string containing this char sequence repeated [n] times.\n * @throws [IllegalArgumentException] when n < 0.\n * @sample samples.text.Strings.repeat\n */\npublic actual fun CharSequence.repeat(n: Int): String {\n    require(n >= 0) { \"Count 'n' must be non-negative, but was $n.\" }\n    return when (n) {\n        0 -> \"\"\n        1 -> this.toString()\n        else -> {\n            var result = \"\"\n            if (!isEmpty()) {\n                var s = this.toString()\n                var count = n\n                while (true) {\n                    if ((count and 1) == 1) {\n                        result += s\n                    }\n                    count = count ushr 1\n                    if (count == 0) {\n                        break\n                    }\n                    s += s\n                }\n            }\n            return result\n        }\n    }\n}\n\n/**\n * Returns a new string obtained by replacing all occurrences of the [oldValue] substring in this string\n * with the specified [newValue] string.\n *\n * @sample samples.text.Strings.replace\n */\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun String.replace(oldValue: String, newValue: String, ignoreCase: Boolean = false): String =\n    nativeReplace(RegExp(Regex.escape(oldValue), if (ignoreCase) \"gui\" else \"gu\"), Regex.nativeEscapeReplacement(newValue))\n\n/**\n * Returns a new string with all occurrences of [oldChar] replaced with [newChar].\n *\n * @sample samples.text.Strings.replace\n */\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun String.replace(oldChar: Char, newChar: Char, ignoreCase: Boolean = false): String =\n    nativeReplace(RegExp(Regex.escape(oldChar.toString()), if (ignoreCase) \"gui\" else \"gu\"), newChar.toString())\n\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun String.replaceFirst(oldValue: String, newValue: String, ignoreCase: Boolean = false): String =\n    nativeReplace(RegExp(Regex.escape(oldValue), if (ignoreCase) \"ui\" else \"u\"), Regex.nativeEscapeReplacement(newValue))\n\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\npublic actual fun String.replaceFirst(oldChar: Char, newChar: Char, ignoreCase: Boolean = false): String =\n    nativeReplace(RegExp(Regex.escape(oldChar.toString()), if (ignoreCase) \"ui\" else \"u\"), newChar.toString())\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n/** Returns the negative [size] if [throwOnMalformed] is false, throws [CharacterCodingException] otherwise. */\nprivate fun malformed(size: Int, index: Int, throwOnMalformed: Boolean): Int {\n    if (throwOnMalformed) throw CharacterCodingException(\"Malformed sequence starting at ${index - 1}\")\n    return -size\n}\n\n/**\n * Returns code point corresponding to UTF-16 surrogate pair,\n * where the first of the pair is the [high] and the second is in the [string] at the [index].\n * Returns zero if the pair is malformed and [throwOnMalformed] is false.\n *\n * @throws CharacterCodingException if the pair is malformed and [throwOnMalformed] is true.\n */\nprivate fun codePointFromSurrogate(string: String, high: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {\n    if (high !in 0xD800..0xDBFF || index >=
```

endIndex) {\n return malformed(0, index, throwOnMalformed)\n }\n val low = string[index].code\n if (low !in 0xDC00..0xDFFF) {\n return malformed(0, index, throwOnMalformed)\n }\n return 0x10000 + ((high and 0x3FF) shl 10) or (low and 0x3FF)\n}\n\n/**\n * Returns code point corresponding to UTF-8 sequence of two bytes,\n * where the first byte of the sequence is the [byte1] and the second byte is in the [bytes] array at the [index].\n * Returns zero if the sequence is malformed and [throwOnMalformed] is false.\n *\n * @throws CharacterCodingException if the sequence of two bytes is malformed and [throwOnMalformed] is true.\n */\nprivate fun codePointFrom2(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {\n if (byte1 and 0x1E == 0 || index >= endIndex) {\n return malformed(0, index, throwOnMalformed)\n }\n val byte2 = bytes[index].toInt()\n if (byte2 and 0xC0 != 0x80) {\n return malformed(0, index, throwOnMalformed)\n }\n return (byte1 shl 6) xor byte2 xor 0xF80\n}\n\n/**\n * Returns code point corresponding to UTF-8 sequence of three bytes,\n * where the first byte of the sequence is the [byte1] and the others are in the [bytes] array starting from the [index].\n * Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence\n * if the sequence is malformed and [throwOnMalformed] is false.\n *\n * @throws CharacterCodingException if the sequence of three bytes is malformed and [throwOnMalformed] is true.\n */\nprivate fun codePointFrom3(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {\n if (index >= endIndex) {\n return malformed(0, index, throwOnMalformed)\n }\n\n val byte2 = bytes[index].toInt()\n if (byte1 and 0xF == 0) {\n if (byte2 and 0xE0 != 0xA0) {\n // Non-shortest form\n return malformed(0, index, throwOnMalformed)\n }\n } else if (byte1 and 0xF == 0xD) {\n if (byte2 and 0xE0 != 0x80) {\n // Surrogate code point\n return malformed(0, index, throwOnMalformed)\n }\n } else if (byte2 and 0xC0 != 0x80) {\n return malformed(0, index, throwOnMalformed)\n }\n\n if (index + 1 == endIndex) {\n return malformed(1, index, throwOnMalformed)\n }\n val byte3 = bytes[index + 1].toInt()\n if (byte3 and 0xC0 != 0x80) {\n return malformed(1, index, throwOnMalformed)\n }\n\n return (byte1 shl 12) xor (byte2 shl 6) xor byte3 xor -0x1E080\n}\n\n/**\n * Returns code point corresponding to UTF-8 sequence of four bytes,\n * where the first byte of the sequence is the [byte1] and the others are in the [bytes] array starting from the [index].\n * Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence\n * if the sequence is malformed and [throwOnMalformed] is false.\n *\n * @throws CharacterCodingException if the sequence of four bytes is malformed and [throwOnMalformed] is true.\n */\nprivate fun codePointFrom4(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {\n if (index >= endIndex) {\n malformed(0, index, throwOnMalformed)\n }\n\n val byte2 = bytes[index].toInt()\n if (byte1 and 0xF == 0x0) {\n if (byte2 and 0xF0 <= 0x80) {\n // Non-shortest form\n return malformed(0, index, throwOnMalformed)\n }\n } else if (byte1 and 0xF == 0x4) {\n if (byte2 and 0xF0 != 0x80) {\n // Out of Unicode code points domain (larger than U+10FFFF)\n return malformed(0, index, throwOnMalformed)\n }\n } else if (byte1 and 0xF > 0x4) {\n return malformed(0, index, throwOnMalformed)\n } else if (byte2 and 0xC0 != 0x80) {\n return malformed(0, index, throwOnMalformed)\n }\n\n if (index + 1 == endIndex) {\n return malformed(1, index, throwOnMalformed)\n }\n val byte3 = bytes[index + 1].toInt()\n if (byte3 and 0xC0 != 0x80) {\n return malformed(1, index, throwOnMalformed)\n }\n\n if (index + 2 == endIndex) {\n return malformed(2, index, throwOnMalformed)\n }\n val byte4 = bytes[index + 2].toInt()\n if (byte4 and 0xC0 != 0x80) {\n return malformed(2, index, throwOnMalformed)\n }\n return (byte1 shl 18) xor (byte2 shl 12) xor (byte3 shl 6) xor byte4 xor 0x381F80\n}\n\n/**\n * Maximum number of bytes needed to encode a single char.\n *\n * Code points in `0..0x7F` are encoded in a single byte.\n * Code points in `0x80..0x7FF` are encoded in two bytes.\n * Code points in `0x800..0xD7FF` or in `0xE000..0xFFFF` are encoded in three bytes.\n * Surrogate code points in `0xD800..0xDFFF` are not Unicode scalar values, therefore aren't encoded.\n * Code points in `0x10000..0x10FFFF` are represented by a pair of surrogate `Char`s and are encoded in four bytes.\n */\nprivate const val MAX_BYTES_PER_CHAR = 3\n\n/**\n * The byte sequence a malformed UTF-16 char sequence is replaced by.\n */\nprivate val REPLACEMENT_BYTE_SEQUENCE: ByteArray = byteArrayOf(0xEF.toByte(), 0xBF.toByte(), 0xBD.toByte())\n\n/**\n * Encodes the [string] using UTF-8 and returns the resulting [ByteArray].\n

```
*\n * @param string the string to encode.\n * @param startIndex the start offset (inclusive) of the substring to encode.\n * @param endIndex the end offset (exclusive) of the substring to encode.\n * @param throwOnMalformed whether to throw on malformed char sequence or replace by the [REPLACEMENT_BYTE_SEQUENCE].\n *\n * @throws CharacterCodingException if the char sequence is malformed and [throwOnMalformed] is true.\n */\ninternal fun encodeUtf8(string: String, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): ByteArray {\n    require(startIndex >= 0 && endIndex <= string.length && startIndex <= endIndex)\n\n    val bytes = ByteArray((endIndex - startIndex) * MAX_BYTES_PER_CHAR)\n    var byteIndex = 0\n    var charIndex = startIndex\n\n    while (charIndex < endIndex) {\n        val code = string[charIndex++].code\n        when {\n            code < 0x80 ->\n                bytes[byteIndex++] = code.toByte()\n            code < 0x800 -> {\n                bytes[byteIndex++] = ((code shr 6) or 0xC0).toByte()\n                bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()\n            }\n            code < 0xD800 || code >= 0xE000 -> {\n                bytes[byteIndex++] = ((code shr 12) or 0xE0).toByte()\n                bytes[byteIndex++] = (((code shr 6) and 0x3F) or 0x80).toByte()\n                bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()\n            }\n            else -> { // Surrogate char value\n                val codePoint = codePointFromSurrogate(string, code, charIndex, endIndex, throwOnMalformed)\n                if (codePoint <= 0) {\n                    bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[0]\n                    bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[1]\n                    bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[2]\n                } else {\n                    bytes[byteIndex++] = ((codePoint shr 18) or 0xF0).toByte()\n                    bytes[byteIndex++] = (((codePoint shr 12) and 0x3F) or 0x80).toByte()\n                    bytes[byteIndex++] = (((codePoint shr 6) and 0x3F) or 0x80).toByte()\n                    bytes[byteIndex++] = ((codePoint and 0x3F) or 0x80).toByte()\n                    charIndex++\n                }\n            }\n        }\n    }\n\n    return if (bytes.size == byteIndex) bytes else bytes.copyOf(byteIndex)\n}\n\n/**\n * The character a malformed UTF-8 byte sequence is replaced by.\n */\nprivate const val REPLACEMENT_CHAR = '\\uFFFD'\n\n/**\n * Decodes the UTF-8 [bytes] array and returns the resulting [String].\n *\n * @param bytes the byte array to decode.\n * @param startIndex the start offset (inclusive) of the array to be decoded.\n * @param endIndex the end offset (exclusive) of the array to be encoded.\n * @param throwOnMalformed whether to throw on malformed byte sequence or replace by the [REPLACEMENT_CHAR].\n *\n * @throws CharacterCodingException if the array is malformed UTF-8 byte sequence and [throwOnMalformed] is true.\n */\ninternal fun decodeUtf8(bytes: ByteArray, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): String {\n    require(startIndex >= 0 && endIndex <= bytes.size && startIndex <= endIndex)\n\n    var byteIndex = startIndex\n    val stringBuilder = StringBuilder()\n\n    while (byteIndex < endIndex) {\n        val byte = bytes[byteIndex++].toInt()\n        when {\n            byte >= 0 ->\n                stringBuilder.append(byte.toChar())\n            byte shr 5 == -2 -> {\n                val code = codePointFrom2(bytes, byte, byteIndex, endIndex, throwOnMalformed)\n                if (code <= 0) {\n                    stringBuilder.append(REPLACEMENT_CHAR)\n                    byteIndex += -code\n                } else {\n                    stringBuilder.append(code.toChar())\n                    byteIndex += 1\n                }\n            }\n            byte shr 4 == -2 -> {\n                val code = codePointFrom3(bytes, byte, byteIndex, endIndex, throwOnMalformed)\n                if (code <= 0) {\n                    stringBuilder.append(REPLACEMENT_CHAR)\n                    byteIndex += -code\n                } else {\n                    stringBuilder.append(code.toChar())\n                    byteIndex += 2\n                }\n            }\n            byte shr 3 == -2 -> {\n                val code = codePointFrom4(bytes, byte, byteIndex, endIndex, throwOnMalformed)\n                if (code <= 0) {\n                    stringBuilder.append(REPLACEMENT_CHAR)\n                    byteIndex += -code\n                } else {\n                    val high = (code - 0x10000) shr 10 or 0xD800\n                    val low = (code and 0x3FF) or 0xDC00\n                    stringBuilder.append(high.toChar())\n                    stringBuilder.append(low.toChar())\n                    byteIndex += 3\n                }\n            }\n            else -> {\n                malformed(0, byteIndex, throwOnMalformed)\n                stringBuilder.append(REPLACEMENT_CHAR)\n            }\n        }\n    }\n\n    return stringBuilder.toString()\n}","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\n/**\n * Returns the detailed description of this throwable with its stack trace.\n *\n * The detailed description includes:\n * - the short description (see
```

[Throwable.toString]) of this throwable;\n * - the complete stack trace;\n * - detailed descriptions of the exceptions that were [suppressed][suppressedExceptions] in order to deliver this exception;\n * - the detailed description of each throwable in the [Throwable.cause] chain.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun Throwable.stackTraceToString(): String = ExceptionTraceBuilder().buildFor(this)\n\n/**\n * Prints the [detailed description][Throwable.stackTraceToString] of this throwable to console error output.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun Throwable.printStackTrace() {\n console.error(this.stackTraceToString())\n}\n\n/**\n * Adds the specified exception to the list of exceptions that were\n * suppressed in order to deliver this exception.\n */\n@SinceKotlin(\"1.4\")\npublic actual fun Throwable.addSuppressed(exception: Throwable) {\n if (this !== exception) {\n val suppressed = this.asDynamic()._suppressed.unsafeCast<MutableList<Throwable>?>()\n if (suppressed == null) {\n this.asDynamic()._suppressed = mutableListOf(exception)\n } else {\n suppressed.add(exception)\n }\n }\n}\n\n/**\n * Returns a list of all exceptions that were suppressed in order to deliver this exception.\n */\n@SinceKotlin(\"1.4\")\npublic actual val Throwable.suppressedExceptions: List<Throwable>\n get() {\n return this.asDynamic()._suppressed?.unsafeCast<List<Throwable>>() ?: emptyList()\n }\n\n\nprivate class ExceptionTraceBuilder {\n private val target = StringBuilder()\n private val visited = arrayOf<Throwable>()\n private var topStack: String = \"\"\n private var topStackStart: Int = 0\n\n fun buildFor(exception: Throwable): String {\n exception.dumpFullTrace(\"\", \"\")\n return target.toString()\n }\n\n private fun hasSeen(exception: Throwable): Boolean = visited.any { it === exception }\n\n private fun Throwable.dumpFullTrace(indent: String, qualifier: String) {\n this.dumpSelfTrace(indent, qualifier) || return\n\n var cause = this.cause\n while (cause != null) {\n cause.dumpSelfTrace(indent, \"Caused by: \") || return\n cause = cause.cause\n }\n }\n\n private fun Throwable.dumpSelfTrace(indent: String, qualifier: String): Boolean {\n target.append(indent).append(qualifier)\n val shortInfo = this.toString()\n if (hasSeen(this)) {\n target.append(\"[CIRCULAR REFERENCE, SEE ABOVE: \").append(shortInfo).append(\"]\\n\")\n return false\n }\n visited.asDynamic().push(this)\n\n var stack = this.asDynamic().stack as String?\n if (stack != null) {\n val stackStart = stack.indexOf(shortInfo).let { if (it < 0) 0 else it + shortInfo.length }\n if (stackStart == 0) target.append(shortInfo).append(\"\\n\")\n if (topStack.isEmpty()) {\n topStack = stack\n topStackStart = stackStart\n } else {\n stack = dropCommonFrames(stack, stackStart)\n }\n if (indent.isNotEmpty()) {\n // indent stack, but avoid indenting exception message lines\n val messageLines = if (stackStart == 0) 0 else 1 + shortInfo.count { c -> c == '\\n' }\n stack.lineSequence().forEachIndexed { index: Int, line: String ->\n if (index >= messageLines) target.append(indent)\n target.append(line).append(\"\\n\")\n }\n } else {\n target.append(stack).append(\"\\n\")\n }\n } else {\n target.append(shortInfo).append(\"\\n\")\n }\n\n val suppressed = suppressedExceptions\n if (suppressed.isNotEmpty()) {\n val suppressedIndent = indent + \" \"\n for (s in suppressed) {\n s.dumpFullTrace(suppressedIndent, \"Suppressed: \")\n }\n }\n return true\n }\n\n private fun dropCommonFrames(stack: String, stackStart: Int): String {\n var commonFrames: Int = 0\n var lastBreak: Int = 0\n var preLastBreak: Int = 0\n for (pos in 0 until minOf(topStack.length - topStackStart, stack.length - stackStart)) {\n val c = stack[stack.lastIndex - pos]\n if (c != topStack[topStack.lastIndex - pos]) break\n if (c == '\\n') {\n commonFrames += 1\n preLastBreak = lastBreak\n lastBreak = pos\n }\n }\n if (commonFrames <= 1) return stack\n while (preLastBreak > 0 && stack[stack.lastIndex - (preLastBreak - 1)] == ' ')\n preLastBreak -= 1\n\n // leave 1 common frame to ease matching with the top exception stack\n return stack.dropLast(preLastBreak) + \"... and ${commonFrames - 1} more common stack frames skipped\"\n }\n}","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.time\n\nimport kotlin.js.json\nimport kotlin.math.*\n\ninternal actual inline val durationAssertionsEnabled: Boolean get() = true\n\ninternal actual fun formatToExactDecimals(value: Double, decimals: Int): String {\n val rounded = if (decimals == 0) {\n value\n } else {\n val pow =

10.0.pow(decimals)\n JsMath.round(abs(value) * pow) / pow * sign(value)\n }\n return if (abs(rounded) < 1e21) {\n // toFixed switches to scientific format after 1e21\n rounded.asDynamic().toFixed(decimals).unsafeCast<String>()\n } else {\n // toPrecision outputs the specified number of digits, but only for positive numbers\n val positive = abs(rounded)\n val positiveString = positive.asDynamic().toPrecision(ceil(log10(positive)) + decimals).unsafeCast<String>()\n if (rounded < 0) \"-$positiveString\" else positiveString\n }\n}\n\ninternal actual fun formatUpToDecimals(value: Double, decimals: Int): String {\n return value.asDynamic().toLocaleString(\"en-us\", json(\"maximumFractionDigits\" to decimals)).unsafeCast<String>()\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.time\n\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalTime::class)\npublic actual enum class DurationUnit(internal val scale: Double) {\n /**\n * Time unit representing one nanosecond, which is 1/1000 of a microsecond.\n */\n NANOSECONDS(1e0),\n /**\n * Time unit representing one microsecond, which is 1/1000 of a millisecond.\n */\n MICROSECONDS(1e3),\n /**\n * Time unit representing one millisecond, which is 1/1000 of a second.\n */\n MILLISECONDS(1e6),\n /**\n * Time unit representing one second.\n */\n SECONDS(1e9),\n /**\n * Time unit representing one minute.\n */\n MINUTES(60e9),\n /**\n * Time unit representing one hour.\n */\n HOURS(3600e9),\n /**\n * Time unit representing one day, which is always equal to 24 hours.\n */\n DAYS(86400e9);\n}\n\n@SinceKotlin(\"1.3\")\ninternal actual fun convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double {\n val sourceCompareTarget = sourceUnit.scale.compareTo(targetUnit.scale)\n return when {\n sourceCompareTarget > 0 -> value * (sourceUnit.scale / targetUnit.scale)\n sourceCompareTarget < 0 -> value / (targetUnit.scale / sourceUnit.scale)\n else -> value\n }\n}\n\n@SinceKotlin(\"1.5\")\ninternal actual fun convertDurationUnitOverflow(value: Long, sourceUnit: DurationUnit, targetUnit: DurationUnit): Long {\n val sourceCompareTarget = sourceUnit.scale.compareTo(targetUnit.scale)\n return when {\n sourceCompareTarget > 0 -> value * (sourceUnit.scale / targetUnit.scale).toLong()\n sourceCompareTarget < 0 -> value / (targetUnit.scale / sourceUnit.scale).toLong()\n else -> value\n }\n}\n\n@SinceKotlin(\"1.5\")\ninternal actual fun convertDurationUnit(value: Long, sourceUnit: DurationUnit, targetUnit: DurationUnit): Long {\n val sourceCompareTarget = sourceUnit.scale.compareTo(targetUnit.scale)\n return when {\n sourceCompareTarget > 0 -> {\n val scale = (sourceUnit.scale / targetUnit.scale).toLong()\n val result = value * scale\n when {\n result / scale == value -> result\n value > 0 -> Long.MAX_VALUE\n else -> Long.MIN_VALUE\n }\n }\n sourceCompareTarget < 0 -> value / (targetUnit.scale / sourceUnit.scale).toLong()\n else -> value\n }\n}\n\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.time\n\nimport org.w3c.performance.GlobalPerformance\nimport org.w3c.performance.Performance\n\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\ninternal actual object MonotonicTimeSource : TimeSource {\n\n private val actualSource: TimeSource = run {\n val isNode: Boolean = js(\"typeof process !== 'undefined' && process.versions && !!process.versions.node\")\n\n if (isNode)\n HrTimeSource(js(\"process\").unsafeCast<Process>())\n else\n js(\"self\").unsafeCast<GlobalPerformance?>()?.performance?.let(::PerformanceTimeSource)\n ?: DateNowTimeSource\n\n }\n\n override fun markNow(): TimeMark = actualSource.markNow()\n}\n\ninternal external interface Process {\n fun hrtime(time: Array<Double> = definedExternally): Array<Double>\n}\n\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\ninternal class HrTimeSource(val process: Process) : TimeSource {\n\n override fun markNow(): TimeMark = object : TimeMark() {\n val startedAt = process.hrtime()\n override fun elapsedNow(): Duration =\n process.hrtime(startedAt).let { (seconds, nanos) -> seconds.toDuration(DurationUnit.SECONDS) + nanos.toDuration(DurationUnit.NANOSECONDS) }\n }\n\n override fun toString(): String = \"TimeSource(process.hrtime())\"\n}\n\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\ninternal class

PerformanceTimeSource(val performance: Performance) : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {\n override fun read(): Double = performance.now()\n override fun toString(): String = \"TimeSource(self.performance.now())\"\n}\n\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\ninternal object DateNowTimeSource : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {\n override fun read(): Double = kotlin.js.Date.now()\n override fun toString(): String = \"TimeSource(Date.now())\"\n}","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlinx.dom\n\nimport org.w3c.dom.*\nimport kotlin.contracts.*\n\n/**\n * Creates a new element with the specified [name].\n *\n * The element is initialized with the specified [init] function.\n */\n@SinceKotlin(\"1.4\")\npublic fun Document.createElement(name: String, init: Element.() -> Unit): Element {\n contract { callsInPlace(init, InvocationKind.EXACTLY_ONCE) }\n return createElement(name).apply(init)\n}\n\n/**\n * Appends a newly created element with the specified [name] to this element.\n *\n * The element is initialized with the specified [init] function.\n */\n@SinceKotlin(\"1.4\")\npublic fun Element.appendElement(name: String, init: Element.() -> Unit): Element {\n contract { callsInPlace(init, InvocationKind.EXACTLY_ONCE) }\n return ownerDocument!!.createElement(name, init).also { appendChild(it) }\n}\n\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlinx.dom\n\nimport org.w3c.dom.*\n\n/** Returns true if the element has the given CSS class style in its 'class' attribute */\n@SinceKotlin(\"1.4\")\nfun Element.hasClass(cssClass: String): Boolean = className.matches(\"\\\"(^|.*\\s+)$cssClass($|\\s+.*)\"\\\".toRegex())\n\n/**\n * Adds CSS class to element. Has no effect if all specified classes are already in class attribute of the element\n *\n * @return true if at least one class has been added\n */\n@SinceKotlin(\"1.4\")\nfun Element.addClass(vararg cssClasses: String): Boolean {\n val missingClasses = cssClasses.filterNot { hasClass(it) }\n if (missingClasses.isNotEmpty()) {\n val presentClasses = className.trim()\n className = buildString {\n append(presentClasses)\n if (!presentClasses.isEmpty()) {\n append(\" \")\n }\n missingClasses.joinTo(this, \" \")\n }\n return true\n }\n\n return false\n}\n\n/**\n * Removes all [cssClasses] from element. Has no effect if all specified classes are missing in class attribute of the element\n *\n * @return true if at least one class has been removed\n */\n@SinceKotlin(\"1.4\")\nfun Element.removeClass(vararg cssClasses: String): Boolean {\n if (cssClasses.any { hasClass(it) }) {\n val toBeRemoved = cssClasses.toSet()\n className = className.trim().split(\"\\\\s+\".toRegex()).filter { it !in toBeRemoved }.joinToString(\" \")\n return true\n }\n\n return false\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"StringsKt\")\n\npackage kotlin.text\n\n/**\n * Converts the string into a regular expression [Regex] with the default options.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.toRegex(): Regex = Regex(this)\n\n/**\n * Converts the string into a regular expression [Regex] with the specified single [option].\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.toRegex(option: RegexOption): Regex = Regex(this, option)\n\n/**\n * Converts the string into a regular expression [Regex] with the specified set of [options].\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.toRegex(options: Set<RegexOption>): Regex = Regex(this, options)\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlinx.dom\n\nimport org.w3c.dom.*\n\n/**\n * Gets a value indicating whether this node is a TEXT_NODE or a CDATA_SECTION_NODE.\n */\n@SinceKotlin(\"1.4\")\npublic val Node.isText: Boolean\n get() = nodeType == Node.TEXT_NODE || nodeType == Node.CDATA_SECTION_NODE\n\n/**\n * Gets a value indicating whether this node is an [Element].\n */\n@SinceKotlin(\"1.4\")\npublic val Node.isElement: Boolean\n get() = nodeType == Node.ELEMENT_NODE\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the

license/LICENSE.txt file.\n */\n\npackage kotlinx.dom\n\nimport org.w3c.dom.*\n\n/** Removes all the children from this node. */\n@SinceKotlin(\"1.4\")\npublic fun Node.clear() {\n while (hasChildNodes()) {\n removeChild(firstChild!!)\n }\n}\n\n/**\n * Creates text node and append it to the element.\n *\n * @return this element\n */\n@SinceKotlin(\"1.4\")\nfun Element.appendText(text: String): Element {\n appendChild(ownerDocument!!.createTextNode(text))\n return this\n}\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage org.w3c.dom\n\n@Deprecated(\"Use UnionMessagePortOrWindowProxy instead.\", ReplaceWith(\"UnionMessagePortOrWindowProxy\"))\ntypealias UnionMessagePortOrWindow = UnionMessagePortOrWindowProxy\n\n@Deprecated(\"Use `as` instead.\", ReplaceWith(\"`as`\"))\nvar HTMLLinkElement.as_\n get() = `as`\n set(value) {\n `as` = value\n }\n\n@Deprecated(\"Use `is` instead.\", ReplaceWith(\"`is`\"))\nvar ElementCreationOptions.is_\n get() = `is`\n set(value) {\n `is` = value\n }","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.khronos.webgl\n\nimport kotlin.js.*\nimport org.w3c.dom.*\nimport org.w3c.dom.events.*\n\npublic external interface WebGLContextAttributes {\n var alpha: Boolean? /* = true */\n get() = definedExternally\n set(value) = definedExternally\n var depth: Boolean? /* = true */\n get() = definedExternally\n set(value) = definedExternally\n var stencil: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n var antialias: Boolean? /* = true */\n get() = definedExternally\n set(value) = definedExternally\n var premultipliedAlpha: Boolean? /* = true */\n get() = definedExternally\n set(value) = definedExternally\n var preserveDrawingBuffer: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n var preferLowPowerToHighPerformance: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n var failIfMajorPerformanceCaveat: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun WebGLContextAttributes(alpha: Boolean? = true, depth: Boolean? = true, stencil: Boolean? = false, antialias: Boolean? = true, premultipliedAlpha: Boolean? = true, preserveDrawingBuffer: Boolean? = false, preferLowPowerToHighPerformance: Boolean? = false, failIfMajorPerformanceCaveat: Boolean? = false): WebGLContextAttributes {\n val o = js(\"({})\")\n o[\"alpha\"] = alpha\n o[\"depth\"] = depth\n o[\"stencil\"] = stencil\n o[\"antialias\"] = antialias\n o[\"premultipliedAlpha\"] = premultipliedAlpha\n o[\"preserveDrawingBuffer\"] = preserveDrawingBuffer\n o[\"preferLowPowerToHighPerformance\"] = preferLowPowerToHighPerformance\n o[\"failIfMajorPerformanceCaveat\"] = failIfMajorPerformanceCaveat\n return o\n}\n\npublic external abstract class WebGLObject\n\n/**\n * Exposes the JavaScript [WebGLBuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLBuffer) to Kotlin\n */\npublic external abstract class WebGLBuffer : WebGLObject\n\n/**\n * Exposes the JavaScript [WebGLFramebuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLFramebuffer) to Kotlin\n */\npublic external abstract class WebGLFramebuffer : WebGLObject\n\n/**\n * Exposes the JavaScript [WebGLProgram](https://developer.mozilla.org/en/docs/Web/API/WebGLProgram) to Kotlin\n */\npublic external abstract class WebGLProgram : WebGLObject\n\n/**\n * Exposes the JavaScript [WebGLRenderbuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLRenderbuffer) to Kotlin\n */\npublic external abstract class WebGLRenderbuffer : WebGLObject\n\n/**\n * Exposes the JavaScript [WebGLShader](https://developer.mozilla.org/en/docs/Web/API/WebGLShader) to Kotlin\n */\npublic external abstract class WebGLShader : WebGLObject\n\n/**\n * Exposes the JavaScript [WebGLTexture](https://developer.mozilla.org/en/docs/Web/API/WebGLTexture) to Kotlin\n */\npublic external abstract class WebGLTexture : WebGLObject\n\n/**\n * Exposes the JavaScript [WebGLUniformLocation](https://developer.mozilla.org/en/docs/Web/API/WebGLUniformLocation) to Kotlin\n */\npublic external abstract class WebGLUniformLocation\n\n/**\n * Exposes the JavaScript

[WebGLActiveInfo](https://developer.mozilla.org/en/docs/Web/API/WebGLActiveInfo) to Kotlin\n */\npublic external abstract class WebGLActiveInfo {\n   open val size: Int\n   open val type: Int\n   open val name: String\n}\n\n/**\n * Exposes the JavaScript [WebGLShaderPrecisionFormat](https://developer.mozilla.org/en/docs/Web/API/WebGLShaderPrecisionFormat) to Kotlin\n */\npublic external abstract class WebGLShaderPrecisionFormat {\n   open val rangeMin: Int\n   open val rangeMax: Int\n   open val precision: Int\n}\n\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface WebGLRenderingContextBase {\n   val canvas: HTMLCanvasElement\n   val drawingBufferWidth: Int\n   val drawingBufferHeight: Int\n   fun getContextAttributes(): WebGLContextAttributes?\n   fun isContextLost(): Boolean\n   fun getSupportedExtensions(): Array<String>?\n   fun getExtension(name: String): dynamic\n   fun activeTexture(texture: Int)\n   fun attachShader(program: WebGLProgram?, shader: WebGLShader?)\n   fun bindAttribLocation(program: WebGLProgram?, index: Int, name: String)\n   fun bindBuffer(target: Int, buffer: WebGLBuffer?)\n   fun bindFramebuffer(target: Int, framebuffer: WebGLFramebuffer?)\n   fun bindRenderbuffer(target: Int, renderbuffer: WebGLRenderbuffer?)\n   fun bindTexture(target: Int, texture: WebGLTexture?)\n   fun blendColor(red: Float, green: Float, blue: Float, alpha: Float)\n   fun blendEquation(mode: Int)\n   fun blendEquationSeparate(modeRGB: Int, modeAlpha: Int)\n   fun blendFunc(sfactor: Int, dfactor: Int)\n   fun blendFuncSeparate(srcRGB: Int, dstRGB: Int, srcAlpha: Int, dstAlpha: Int)\n   fun bufferData(target: Int, size: Int, usage: Int)\n   fun bufferData(target: Int, data: BufferDataSource?, usage: Int)\n   fun bufferSubData(target: Int, offset: Int, data: BufferDataSource?)\n   fun checkFramebufferStatus(target: Int): Int\n   fun clear(mask: Int)\n   fun clearColor(red: Float, green: Float, blue: Float, alpha: Float)\n   fun clearDepth(depth: Float)\n   fun clearStencil(s: Int)\n   fun colorMask(red: Boolean, green: Boolean, blue: Boolean, alpha: Boolean)\n   fun compileShader(shader: WebGLShader?)\n   fun compressedTexImage2D(target: Int, level: Int, internalformat: Int, width: Int, height: Int, border: Int, data: ArrayBufferView)\n   fun compressedTexSubImage2D(target: Int, level: Int, xoffset: Int, yoffset: Int, width: Int, height: Int, format: Int, data: ArrayBufferView)\n   fun copyTexImage2D(target: Int, level: Int, internalformat: Int, x: Int, y: Int, width: Int, height: Int, border: Int)\n   fun copyTexSubImage2D(target: Int, level: Int, xoffset: Int, yoffset: Int, x: Int, y: Int, width: Int, height: Int)\n   fun createBuffer(): WebGLBuffer?\n   fun createFramebuffer(): WebGLFramebuffer?\n   fun createProgram(): WebGLProgram?\n   fun createRenderbuffer(): WebGLRenderbuffer?\n   fun createShader(type: Int): WebGLShader?\n   fun createTexture(): WebGLTexture?\n   fun cullFace(mode: Int)\n   fun deleteBuffer(buffer: WebGLBuffer?)\n   fun deleteFramebuffer(framebuffer: WebGLFramebuffer?)\n   fun deleteProgram(program: WebGLProgram?)\n   fun deleteRenderbuffer(renderbuffer: WebGLRenderbuffer?)\n   fun deleteShader(shader: WebGLShader?)\n   fun deleteTexture(texture: WebGLTexture?)\n   fun depthFunc(func: Int)\n   fun depthMask(flag: Boolean)\n   fun depthRange(zNear: Float, zFar: Float)\n   fun detachShader(program: WebGLProgram?, shader: WebGLShader?)\n   fun disable(cap: Int)\n   fun disableVertexAttribArray(index: Int)\n   fun drawArrays(mode: Int, first: Int, count: Int)\n   fun drawElements(mode: Int, count: Int, type: Int, offset: Int)\n   fun enable(cap: Int)\n   fun enableVertexAttribArray(index: Int)\n   fun finish()\n   fun flush()\n   fun framebufferRenderbuffer(target: Int, attachment: Int, renderbuffertarget: Int, renderbuffer: WebGLRenderbuffer?)\n   fun framebufferTexture2D(target: Int, attachment: Int, textarget: Int, texture: WebGLTexture?, level: Int)\n   fun frontFace(mode: Int)\n   fun generateMipmap(target: Int)\n   fun getActiveAttrib(program: WebGLProgram?, index: Int): WebGLActiveInfo?\n   fun getActiveUniform(program: WebGLProgram?, index: Int): WebGLActiveInfo?\n   fun getAttachedShaders(program: WebGLProgram?): Array<WebGLShader>?\n   fun getAttribLocation(program: WebGLProgram?, name: String): Int\n   fun getBufferParameter(target: Int, pname: Int): Any?\n   fun getParameter(pname: Int): Any?\n   fun getError(): Int\n   fun getFramebufferAttachmentParameter(target: Int, attachment: Int, pname: Int): Any?\n   fun getProgramParameter(program: WebGLProgram?, pname: Int): Any?\n   fun getProgramInfoLog(program: WebGLProgram?): String?\n   fun getRenderbufferParameter(target: Int, pname: Int): Any?\n   fun getShaderParameter(shader: WebGLShader?, pname: Int): Any?\n   fun getShaderPrecisionFormat(shadertype: Int, precisiontype: Int): WebGLShaderPrecisionFormat?\n   fun

getShaderInfoLog(shader: WebGLShader?): String?\n fun getShaderSource(shader: WebGLShader?): String?\n fun getTexParameter(target: Int, pname: Int): Any?\n fun getUniform(program: WebGLProgram?, location: WebGLUniformLocation?): Any?\n fun getUniformLocation(program: WebGLProgram?, name: String): WebGLUniformLocation?\n fun getVertexAttrib(index: Int, pname: Int): Any?\n fun getVertexAttribOffset(index: Int, pname: Int): Int\n fun hint(target: Int, mode: Int)\n fun isBuffer(buffer: WebGLBuffer?): Boolean\n fun isEnabled(cap: Int): Boolean\n fun isFramebuffer(framebuffer: WebGLFramebuffer?): Boolean\n fun isProgram(program: WebGLProgram?): Boolean\n fun isRenderbuffer(renderbuffer: WebGLRenderbuffer?): Boolean\n fun isShader(shader: WebGLShader?): Boolean\n fun isTexture(texture: WebGLTexture?): Boolean\n fun lineWidth(width: Float)\n fun linkProgram(program: WebGLProgram?)\n fun pixelStorei(pname: Int, param: Int)\n fun polygonOffset(factor: Float, units: Float)\n fun readPixels(x: Int, y: Int, width: Int, height: Int, format: Int, type: Int, pixels: ArrayBufferView?)\n fun renderbufferStorage(target: Int, internalformat: Int, width: Int, height: Int)\n fun sampleCoverage(value: Float, invert: Boolean)\n fun scissor(x: Int, y: Int, width: Int, height: Int)\n fun shaderSource(shader: WebGLShader?, source: String)\n fun stencilFunc(func: Int, ref: Int, mask: Int)\n fun stencilFuncSeparate(face: Int, func: Int, ref: Int, mask: Int)\n fun stencilMask(mask: Int)\n fun stencilMaskSeparate(face: Int, mask: Int)\n fun stencilOp(fail: Int, zfail: Int, zpass: Int)\n fun stencilOpSeparate(face: Int, fail: Int, zfail: Int, zpass: Int)\n fun texImage2D(target: Int, level: Int, internalformat: Int, width: Int, height: Int, border: Int, format: Int, type: Int, pixels: ArrayBufferView?)\n fun texImage2D(target: Int, level: Int, internalformat: Int, format: Int, type: Int, source: TexImageSource?)\n fun texParameterf(target: Int, pname: Int, param: Float)\n fun texParameteri(target: Int, pname: Int, param: Int)\n fun texSubImage2D(target: Int, level: Int, xoffset: Int, yoffset: Int, width: Int, height: Int, format: Int, type: Int, pixels: ArrayBufferView?)\n fun texSubImage2D(target: Int, level: Int, xoffset: Int, yoffset: Int, format: Int, type: Int, source: TexImageSource?)\n fun uniform1f(location: WebGLUniformLocation?, x: Float)\n fun uniform1fv(location: WebGLUniformLocation?, v: Float32Array)\n fun uniform1fv(location: WebGLUniformLocation?, v: Array<Float>)\n fun uniform1i(location: WebGLUniformLocation?, x: Int)\n fun uniform1iv(location: WebGLUniformLocation?, v: Int32Array)\n fun uniform1iv(location: WebGLUniformLocation?, v: Array<Int>)\n fun uniform2f(location: WebGLUniformLocation?, x: Float, y: Float)\n fun uniform2fv(location: WebGLUniformLocation?, v: Float32Array)\n fun uniform2fv(location: WebGLUniformLocation?, v: Array<Float>)\n fun uniform2i(location: WebGLUniformLocation?, x: Int, y: Int)\n fun uniform2iv(location: WebGLUniformLocation?, v: Int32Array)\n fun uniform2iv(location: WebGLUniformLocation?, v: Array<Int>)\n fun uniform3f(location: WebGLUniformLocation?, x: Float, y: Float, z: Float)\n fun uniform3fv(location: WebGLUniformLocation?, v: Float32Array)\n fun uniform3fv(location: WebGLUniformLocation?, v: Array<Float>)\n fun uniform3i(location: WebGLUniformLocation?, x: Int, y: Int, z: Int)\n fun uniform3iv(location: WebGLUniformLocation?, v: Int32Array)\n fun uniform3iv(location: WebGLUniformLocation?, v: Array<Int>)\n fun uniform4f(location: WebGLUniformLocation?, x: Float, y: Float, z: Float, w: Float)\n fun uniform4fv(location: WebGLUniformLocation?, v: Float32Array)\n fun uniform4fv(location: WebGLUniformLocation?, v: Array<Float>)\n fun uniform4i(location: WebGLUniformLocation?, x: Int, y: Int, z: Int, w: Int)\n fun uniform4iv(location: WebGLUniformLocation?, v: Int32Array)\n fun uniform4iv(location: WebGLUniformLocation?, v: Array<Int>)\n fun uniformMatrix2fv(location: WebGLUniformLocation?, transpose: Boolean, value: Float32Array)\n fun uniformMatrix2fv(location: WebGLUniformLocation?, transpose: Boolean, value: Array<Float>)\n fun uniformMatrix3fv(location: WebGLUniformLocation?, transpose: Boolean, value: Float32Array)\n fun uniformMatrix3fv(location: WebGLUniformLocation?, transpose: Boolean, value: Array<Float>)\n fun uniformMatrix4fv(location: WebGLUniformLocation?, transpose: Boolean, value: Float32Array)\n fun uniformMatrix4fv(location: WebGLUniformLocation?, transpose: Boolean, value: Array<Float>)\n fun useProgram(program: WebGLProgram?)\n fun validateProgram(program: WebGLProgram?)\n fun vertexAttrib1f(index: Int, x: Float)\n fun vertexAttrib1fv(index: Int, values: dynamic)\n fun vertexAttrib2f(index: Int, x: Float, y: Float)\n fun vertexAttrib2fv(index: Int, values: dynamic)\n fun vertexAttrib3f(index: Int, x: Float, y: Float, z: Float)\n fun vertexAttrib3fv(index: Int, values: dynamic)\n fun

vertexAttrib4f(index: Int, x: Float, y: Float, z: Float, w: Float)\n fun vertexAttrib4fv(index: Int, values: dynamic)\n fun vertexAttribPointer(index: Int, size: Int, type: Int, normalized: Boolean, stride: Int, offset: Int)\n fun viewport(x: Int, y: Int, width: Int, height: Int)\n\n companion object {\n val DEPTH_BUFFER_BIT: Int\n val STENCIL_BUFFER_BIT: Int\n val COLOR_BUFFER_BIT: Int\n val POINTS: Int\n val LINES: Int\n val LINE_LOOP: Int\n val LINE_STRIP: Int\n val TRIANGLES: Int\n val TRIANGLE_STRIP: Int\n val TRIANGLE_FAN: Int\n val ZERO: Int\n val ONE: Int\n val SRC_COLOR: Int\n val ONE_MINUS_SRC_COLOR: Int\n val SRC_ALPHA: Int\n val ONE_MINUS_SRC_ALPHA: Int\n val DST_ALPHA: Int\n val ONE_MINUS_DST_ALPHA: Int\n val DST_COLOR: Int\n val ONE_MINUS_DST_COLOR: Int\n val SRC_ALPHA_SATURATE: Int\n val FUNC_ADD: Int\n val BLEND_EQUATION: Int\n val BLEND_EQUATION_RGB: Int\n val BLEND_EQUATION_ALPHA: Int\n val FUNC_SUBTRACT: Int\n val FUNC_REVERSE_SUBTRACT: Int\n val BLEND_DST_RGB: Int\n val BLEND_SRC_RGB: Int\n val BLEND_DST_ALPHA: Int\n val BLEND_SRC_ALPHA: Int\n val CONSTANT_COLOR: Int\n val ONE_MINUS_CONSTANT_COLOR: Int\n val CONSTANT_ALPHA: Int\n val ONE_MINUS_CONSTANT_ALPHA: Int\n val BLEND_COLOR: Int\n val ARRAY_BUFFER: Int\n val ELEMENT_ARRAY_BUFFER: Int\n val ARRAY_BUFFER_BINDING: Int\n val ELEMENT_ARRAY_BUFFER_BINDING: Int\n val STREAM_DRAW: Int\n val STATIC_DRAW: Int\n val DYNAMIC_DRAW: Int\n val BUFFER_SIZE: Int\n val BUFFER_USAGE: Int\n val CURRENT_VERTEX_ATTRIB: Int\n val FRONT: Int\n val BACK: Int\n val FRONT_AND_BACK: Int\n val CULL_FACE: Int\n val BLEND: Int\n val DITHER: Int\n val STENCIL_TEST: Int\n val DEPTH_TEST: Int\n val SCISSOR_TEST: Int\n val POLYGON_OFFSET_FILL: Int\n val SAMPLE_ALPHA_TO_COVERAGE: Int\n val SAMPLE_COVERAGE: Int\n val NO_ERROR: Int\n val INVALID_ENUM: Int\n val INVALID_VALUE: Int\n val INVALID_OPERATION: Int\n val OUT_OF_MEMORY: Int\n val CW: Int\n val CCW: Int\n val LINE_WIDTH: Int\n val ALIASED_POINT_SIZE_RANGE: Int\n val ALIASED_LINE_WIDTH_RANGE: Int\n val CULL_FACE_MODE: Int\n val FRONT_FACE: Int\n val DEPTH_RANGE: Int\n val DEPTH_WRITEMASK: Int\n val DEPTH_CLEAR_VALUE: Int\n val DEPTH_FUNC: Int\n val STENCIL_CLEAR_VALUE: Int\n val STENCIL_FUNC: Int\n val STENCIL_FAIL: Int\n val STENCIL_PASS_DEPTH_FAIL: Int\n val STENCIL_PASS_DEPTH_PASS: Int\n val STENCIL_REF: Int\n val STENCIL_VALUE_MASK: Int\n val STENCIL_WRITEMASK: Int\n val STENCIL_BACK_FUNC: Int\n val STENCIL_BACK_FAIL: Int\n val STENCIL_BACK_PASS_DEPTH_FAIL: Int\n val STENCIL_BACK_PASS_DEPTH_PASS: Int\n val STENCIL_BACK_REF: Int\n val STENCIL_BACK_VALUE_MASK: Int\n val STENCIL_BACK_WRITEMASK: Int\n val VIEWPORT: Int\n val SCISSOR_BOX: Int\n val COLOR_CLEAR_VALUE: Int\n val COLOR_WRITEMASK: Int\n val UNPACK_ALIGNMENT: Int\n val PACK_ALIGNMENT: Int\n val MAX_TEXTURE_SIZE: Int\n val MAX_VIEWPORT_DIMS: Int\n val SUBPIXEL_BITS: Int\n val RED_BITS: Int\n val GREEN_BITS: Int\n val BLUE_BITS: Int\n val ALPHA_BITS: Int\n val DEPTH_BITS: Int\n val STENCIL_BITS: Int\n val POLYGON_OFFSET_UNITS: Int\n val POLYGON_OFFSET_FACTOR: Int\n val TEXTURE_BINDING_2D: Int\n val SAMPLE_BUFFERS: Int\n val SAMPLES: Int\n val SAMPLE_COVERAGE_VALUE: Int\n val SAMPLE_COVERAGE_INVERT: Int\n val COMPRESSED_TEXTURE_FORMATS: Int\n val DONT_CARE: Int\n val FASTEST: Int\n val NICEST: Int\n val GENERATE_MIPMAP_HINT: Int\n val BYTE: Int\n val UNSIGNED_BYTE: Int\n val SHORT: Int\n val UNSIGNED_SHORT: Int\n val INT: Int\n val UNSIGNED_INT: Int\n val FLOAT: Int\n val DEPTH_COMPONENT: Int\n val ALPHA: Int\n val RGB: Int\n val RGBA: Int\n val LUMINANCE: Int\n val LUMINANCE_ALPHA: Int\n val UNSIGNED_SHORT_4_4_4_4: Int\n val UNSIGNED_SHORT_5_5_5_1: Int\n val UNSIGNED_SHORT_5_6_5: Int\n val FRAGMENT_SHADER: Int\n val VERTEX_SHADER: Int\n

val MAX_VERTEX_ATTRIBS: Int\n val MAX_VERTEX_UNIFORM_VECTORS: Int\n val MAX_VARYING_VECTORS: Int\n val MAX_COMBINED_TEXTURE_IMAGE_UNITS: Int\n val MAX_VERTEX_TEXTURE_IMAGE_UNITS: Int\n val MAX_TEXTURE_IMAGE_UNITS: Int\n val MAX_FRAGMENT_UNIFORM_VECTORS: Int\n val SHADER_TYPE: Int\n val DELETE_STATUS: Int\n val LINK_STATUS: Int\n val VALIDATE_STATUS: Int\n val ATTACHED_SHADERS: Int\n val ACTIVE_UNIFORMS: Int\n val ACTIVE_ATTRIBUTES: Int\n val SHADING_LANGUAGE_VERSION: Int\n val CURRENT_PROGRAM: Int\n val NEVER: Int\n val LESS: Int\n val EQUAL: Int\n val LEQUAL: Int\n val GREATER: Int\n val NOTEQUAL: Int\n val GEQUAL: Int\n val ALWAYS: Int\n val KEEP: Int\n val REPLACE: Int\n val INCR: Int\n val DECR: Int\n val INVERT: Int\n val INCR_WRAP: Int\n val DECR_WRAP: Int\n val VENDOR: Int\n val RENDERER: Int\n val VERSION: Int\n val NEAREST: Int\n val LINEAR: Int\n val NEAREST_MIPMAP_NEAREST: Int\n val LINEAR_MIPMAP_NEAREST: Int\n val NEAREST_MIPMAP_LINEAR: Int\n val LINEAR_MIPMAP_LINEAR: Int\n val TEXTURE_MAG_FILTER: Int\n val TEXTURE_MIN_FILTER: Int\n val TEXTURE_WRAP_S: Int\n val TEXTURE_WRAP_T: Int\n val TEXTURE_2D: Int\n val TEXTURE: Int\n val TEXTURE_CUBE_MAP: Int\n val TEXTURE_BINDING_CUBE_MAP: Int\n val TEXTURE_CUBE_MAP_POSITIVE_X: Int\n val TEXTURE_CUBE_MAP_NEGATIVE_X: Int\n val TEXTURE_CUBE_MAP_POSITIVE_Y: Int\n val TEXTURE_CUBE_MAP_NEGATIVE_Y: Int\n val TEXTURE_CUBE_MAP_POSITIVE_Z: Int\n val TEXTURE_CUBE_MAP_NEGATIVE_Z: Int\n val MAX_CUBE_MAP_TEXTURE_SIZE: Int\n val TEXTURE0: Int\n val TEXTURE1: Int\n val TEXTURE2: Int\n val TEXTURE3: Int\n val TEXTURE4: Int\n val TEXTURE5: Int\n val TEXTURE6: Int\n val TEXTURE7: Int\n val TEXTURE8: Int\n val TEXTURE9: Int\n val TEXTURE10: Int\n val TEXTURE11: Int\n val TEXTURE12: Int\n val TEXTURE13: Int\n val TEXTURE14: Int\n val TEXTURE15: Int\n val TEXTURE16: Int\n val TEXTURE17: Int\n val TEXTURE18: Int\n val TEXTURE19: Int\n val TEXTURE20: Int\n val TEXTURE21: Int\n val TEXTURE22: Int\n val TEXTURE23: Int\n val TEXTURE24: Int\n val TEXTURE25: Int\n val TEXTURE26: Int\n val TEXTURE27: Int\n val TEXTURE28: Int\n val TEXTURE29: Int\n val TEXTURE30: Int\n val TEXTURE31: Int\n val ACTIVE_TEXTURE: Int\n val REPEAT: Int\n val CLAMP_TO_EDGE: Int\n val MIRRORED_REPEAT: Int\n val FLOAT_VEC2: Int\n val FLOAT_VEC3: Int\n val FLOAT_VEC4: Int\n val INT_VEC2: Int\n val INT_VEC3: Int\n val INT_VEC4: Int\n val BOOL: Int\n val BOOL_VEC2: Int\n val BOOL_VEC3: Int\n val BOOL_VEC4: Int\n val FLOAT_MAT2: Int\n val FLOAT_MAT3: Int\n val FLOAT_MAT4: Int\n val SAMPLER_2D: Int\n val SAMPLER_CUBE: Int\n val VERTEX_ATTRIB_ARRAY_ENABLED: Int\n val VERTEX_ATTRIB_ARRAY_SIZE: Int\n val VERTEX_ATTRIB_ARRAY_STRIDE: Int\n val VERTEX_ATTRIB_ARRAY_TYPE: Int\n val VERTEX_ATTRIB_ARRAY_NORMALIZED: Int\n val VERTEX_ATTRIB_ARRAY_POINTER: Int\n val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int\n val IMPLEMENTATION_COLOR_READ_TYPE: Int\n val IMPLEMENTATION_COLOR_READ_FORMAT: Int\n val COMPILE_STATUS: Int\n val LOW_FLOAT: Int\n val MEDIUM_FLOAT: Int\n val HIGH_FLOAT: Int\n val LOW_INT: Int\n val MEDIUM_INT: Int\n val HIGH_INT: Int\n val FRAMEBUFFER: Int\n val RENDERBUFFER: Int\n val RGBA4: Int\n val RGB5_A1: Int\n val RGB565: Int\n val DEPTH_COMPONENT16: Int\n val STENCIL_INDEX: Int\n val STENCIL_INDEX8: Int\n val DEPTH_STENCIL: Int\n val RENDERBUFFER_WIDTH: Int\n val RENDERBUFFER_HEIGHT: Int\n val RENDERBUFFER_INTERNAL_FORMAT: Int\n val RENDERBUFFER_RED_SIZE: Int\n val RENDERBUFFER_GREEN_SIZE: Int\n val RENDERBUFFER_BLUE_SIZE: Int\n val RENDERBUFFER_ALPHA_SIZE: Int\n val RENDERBUFFER_DEPTH_SIZE: Int\n val RENDERBUFFER_STENCIL_SIZE: Int\n val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int\n val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int\n val

FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int\n val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int\n val COLOR_ATTACHMENT0: Int\n val DEPTH_ATTACHMENT: Int\n val STENCIL_ATTACHMENT: Int\n val DEPTH_STENCIL_ATTACHMENT: Int\n val NONE: Int\n val FRAMEBUFFER_COMPLETE: Int\n val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int\n val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int\n val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int\n val FRAMEBUFFER_UNSUPPORTED: Int\n val FRAMEBUFFER_BINDING: Int\n val RENDERBUFFER_BINDING: Int\n val MAX_RENDERBUFFER_SIZE: Int\n val INVALID_FRAMEBUFFER_OPERATION: Int\n val UNPACK_FLIP_Y_WEBGL: Int\n val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int\n val CONTEXT_LOST_WEBGL: Int\n val UNPACK_COLORSPACE_CONVERSION_WEBGL: Int\n val BROWSER_DEFAULT_WEBGL: Int\n }\n}\n\n/**\n * Exposes the JavaScript [WebGLRenderingContext](https://developer.mozilla.org/en/docs/Web/API/WebGLRenderingContext) to Kotlin\n */\npublic external abstract class WebGLRenderingContext : WebGLRenderingContextBase, RenderingContext {\n companion object {\n val DEPTH_BUFFER_BIT: Int\n val STENCIL_BUFFER_BIT: Int\n val COLOR_BUFFER_BIT: Int\n val POINTS: Int\n val LINES: Int\n val LINE_LOOP: Int\n val LINE_STRIP: Int\n val TRIANGLES: Int\n val TRIANGLE_STRIP: Int\n val TRIANGLE_FAN: Int\n val ZERO: Int\n val ONE: Int\n val SRC_COLOR: Int\n val ONE_MINUS_SRC_COLOR: Int\n val SRC_ALPHA: Int\n val ONE_MINUS_SRC_ALPHA: Int\n val DST_ALPHA: Int\n val ONE_MINUS_DST_ALPHA: Int\n val DST_COLOR: Int\n val ONE_MINUS_DST_COLOR: Int\n val SRC_ALPHA_SATURATE: Int\n val FUNC_ADD: Int\n val BLEND_EQUATION: Int\n val BLEND_EQUATION_RGB: Int\n val BLEND_EQUATION_ALPHA: Int\n val FUNC_SUBTRACT: Int\n val FUNC_REVERSE_SUBTRACT: Int\n val BLEND_DST_RGB: Int\n val BLEND_SRC_RGB: Int\n val BLEND_DST_ALPHA: Int\n val BLEND_SRC_ALPHA: Int\n val CONSTANT_COLOR: Int\n val ONE_MINUS_CONSTANT_COLOR: Int\n val CONSTANT_ALPHA: Int\n val ONE_MINUS_CONSTANT_ALPHA: Int\n val BLEND_COLOR: Int\n val ARRAY_BUFFER: Int\n val ELEMENT_ARRAY_BUFFER: Int\n val ARRAY_BUFFER_BINDING: Int\n val ELEMENT_ARRAY_BUFFER_BINDING: Int\n val STREAM_DRAW: Int\n val STATIC_DRAW: Int\n val DYNAMIC_DRAW: Int\n val BUFFER_SIZE: Int\n val BUFFER_USAGE: Int\n val CURRENT_VERTEX_ATTRIB: Int\n val FRONT: Int\n val BACK: Int\n val FRONT_AND_BACK: Int\n val CULL_FACE: Int\n val BLEND: Int\n val DITHER: Int\n val STENCIL_TEST: Int\n val DEPTH_TEST: Int\n val SCISSOR_TEST: Int\n val POLYGON_OFFSET_FILL: Int\n val SAMPLE_ALPHA_TO_COVERAGE: Int\n val SAMPLE_COVERAGE: Int\n val NO_ERROR: Int\n val INVALID_ENUM: Int\n val INVALID_VALUE: Int\n val INVALID_OPERATION: Int\n val OUT_OF_MEMORY: Int\n val CW: Int\n val CCW: Int\n val LINE_WIDTH: Int\n val ALIASED_POINT_SIZE_RANGE: Int\n val ALIASED_LINE_WIDTH_RANGE: Int\n val CULL_FACE_MODE: Int\n val FRONT_FACE: Int\n val DEPTH_RANGE: Int\n val DEPTH_WRITEMASK: Int\n val DEPTH_CLEAR_VALUE: Int\n val DEPTH_FUNC: Int\n val STENCIL_CLEAR_VALUE: Int\n val STENCIL_FUNC: Int\n val STENCIL_FAIL: Int\n val STENCIL_PASS_DEPTH_FAIL: Int\n val STENCIL_PASS_DEPTH_PASS: Int\n val STENCIL_REF: Int\n val STENCIL_VALUE_MASK: Int\n val STENCIL_WRITEMASK: Int\n val STENCIL_BACK_FUNC: Int\n val STENCIL_BACK_FAIL: Int\n val STENCIL_BACK_PASS_DEPTH_FAIL: Int\n val STENCIL_BACK_PASS_DEPTH_PASS: Int\n val STENCIL_BACK_REF: Int\n val STENCIL_BACK_VALUE_MASK: Int\n val STENCIL_BACK_WRITEMASK: Int\n val VIEWPORT: Int\n val SCISSOR_BOX: Int\n val COLOR_CLEAR_VALUE: Int\n val COLOR_WRITEMASK: Int\n val UNPACK_ALIGNMENT: Int\n val PACK_ALIGNMENT: Int\n val MAX_TEXTURE_SIZE: Int\n val MAX_VIEWPORT_DIMS: Int\n val SUBPIXEL_BITS: Int\n val RED_BITS: Int\n val GREEN_BITS: Int\n val BLUE_BITS: Int\n

val ALPHA_BITS: Int\n val DEPTH_BITS: Int\n val STENCIL_BITS: Int\n val POLYGON_OFFSET_UNITS: Int\n val POLYGON_OFFSET_FACTOR: Int\n val TEXTURE_BINDING_2D: Int\n val SAMPLE_BUFFERS: Int\n val SAMPLES: Int\n val SAMPLE_COVERAGE_VALUE: Int\n val SAMPLE_COVERAGE_INVERT: Int\n val COMPRESSED_TEXTURE_FORMATS: Int\n val DONT_CARE: Int\n val FASTEST: Int\n val NICEST: Int\n val GENERATE_MIPMAP_HINT: Int\n val BYTE: Int\n val UNSIGNED_BYTE: Int\n val SHORT: Int\n val UNSIGNED_SHORT: Int\n val INT: Int\n val UNSIGNED_INT: Int\n val FLOAT: Int\n val DEPTH_COMPONENT: Int\n val ALPHA: Int\n val RGB: Int\n val RGBA: Int\n val LUMINANCE: Int\n val LUMINANCE_ALPHA: Int\n val UNSIGNED_SHORT_4_4_4_4: Int\n val UNSIGNED_SHORT_5_5_5_1: Int\n val UNSIGNED_SHORT_5_6_5: Int\n val FRAGMENT_SHADER: Int\n val VERTEX_SHADER: Int\n val MAX_VERTEX_ATTRIBS: Int\n val MAX_VERTEX_UNIFORM_VECTORS: Int\n val MAX_VARYING_VECTORS: Int\n val MAX_COMBINED_TEXTURE_IMAGE_UNITS: Int\n val MAX_VERTEX_TEXTURE_IMAGE_UNITS: Int\n val MAX_TEXTURE_IMAGE_UNITS: Int\n val MAX_FRAGMENT_UNIFORM_VECTORS: Int\n val SHADER_TYPE: Int\n val DELETE_STATUS: Int\n val LINK_STATUS: Int\n val VALIDATE_STATUS: Int\n val ATTACHED_SHADERS: Int\n val ACTIVE_UNIFORMS: Int\n val ACTIVE_ATTRIBUTES: Int\n val SHADING_LANGUAGE_VERSION: Int\n val CURRENT_PROGRAM: Int\n val NEVER: Int\n val LESS: Int\n val EQUAL: Int\n val LEQUAL: Int\n val GREATER: Int\n val NOTEQUAL: Int\n val GEQUAL: Int\n val ALWAYS: Int\n val KEEP: Int\n val REPLACE: Int\n val INCR: Int\n val DECR: Int\n val INVERT: Int\n val INCR_WRAP: Int\n val DECR_WRAP: Int\n val VENDOR: Int\n val RENDERER: Int\n val VERSION: Int\n val NEAREST: Int\n val LINEAR: Int\n val NEAREST_MIPMAP_NEAREST: Int\n val LINEAR_MIPMAP_NEAREST: Int\n val NEAREST_MIPMAP_LINEAR: Int\n val LINEAR_MIPMAP_LINEAR: Int\n val TEXTURE_MAG_FILTER: Int\n val TEXTURE_MIN_FILTER: Int\n val TEXTURE_WRAP_S: Int\n val TEXTURE_WRAP_T: Int\n val TEXTURE_2D: Int\n val TEXTURE: Int\n val TEXTURE_CUBE_MAP: Int\n val TEXTURE_BINDING_CUBE_MAP: Int\n val TEXTURE_CUBE_MAP_POSITIVE_X: Int\n val TEXTURE_CUBE_MAP_NEGATIVE_X: Int\n val TEXTURE_CUBE_MAP_POSITIVE_Y: Int\n val TEXTURE_CUBE_MAP_NEGATIVE_Y: Int\n val TEXTURE_CUBE_MAP_POSITIVE_Z: Int\n val TEXTURE_CUBE_MAP_NEGATIVE_Z: Int\n val MAX_CUBE_MAP_TEXTURE_SIZE: Int\n val TEXTURE0: Int\n val TEXTURE1: Int\n val TEXTURE2: Int\n val TEXTURE3: Int\n val TEXTURE4: Int\n val TEXTURE5: Int\n val TEXTURE6: Int\n val TEXTURE7: Int\n val TEXTURE8: Int\n val TEXTURE9: Int\n val TEXTURE10: Int\n val TEXTURE11: Int\n val TEXTURE12: Int\n val TEXTURE13: Int\n val TEXTURE14: Int\n val TEXTURE15: Int\n val TEXTURE16: Int\n val TEXTURE17: Int\n val TEXTURE18: Int\n val TEXTURE19: Int\n val TEXTURE20: Int\n val TEXTURE21: Int\n val TEXTURE22: Int\n val TEXTURE23: Int\n val TEXTURE24: Int\n val TEXTURE25: Int\n val TEXTURE26: Int\n val TEXTURE27: Int\n val TEXTURE28: Int\n val TEXTURE29: Int\n val TEXTURE30: Int\n val TEXTURE31: Int\n val ACTIVE_TEXTURE: Int\n val REPEAT: Int\n val CLAMP_TO_EDGE: Int\n val MIRRORED_REPEAT: Int\n val FLOAT_VEC2: Int\n val FLOAT_VEC3: Int\n val FLOAT_VEC4: Int\n val INT_VEC2: Int\n val INT_VEC3: Int\n val INT_VEC4: Int\n val BOOL: Int\n val BOOL_VEC2: Int\n val BOOL_VEC3: Int\n val BOOL_VEC4: Int\n val FLOAT_MAT2: Int\n val FLOAT_MAT3: Int\n val FLOAT_MAT4: Int\n val SAMPLER_2D: Int\n val SAMPLER_CUBE: Int\n val VERTEX_ATTRIB_ARRAY_ENABLED: Int\n val VERTEX_ATTRIB_ARRAY_SIZE: Int\n val VERTEX_ATTRIB_ARRAY_STRIDE: Int\n val VERTEX_ATTRIB_ARRAY_TYPE: Int\n val VERTEX_ATTRIB_ARRAY_NORMALIZED: Int\n val VERTEX_ATTRIB_ARRAY_POINTER: Int\n val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int\n val IMPLEMENTATION_COLOR_READ_TYPE: Int\n val

IMPLEMENTATION_COLOR_READ_FORMAT: Int\n val COMPILE_STATUS: Int\n val LOW_FLOAT: Int\n val MEDIUM_FLOAT: Int\n val HIGH_FLOAT: Int\n val LOW_INT: Int\n val MEDIUM_INT: Int\n val HIGH_INT: Int\n val FRAMEBUFFER: Int\n val RENDERBUFFER: Int\n val RGBA4: Int\n val RGB5_A1: Int\n val RGB565: Int\n val DEPTH_COMPONENT16: Int\n val STENCIL_INDEX: Int\n val STENCIL_INDEX8: Int\n val DEPTH_STENCIL: Int\n val RENDERBUFFER_WIDTH: Int\n val RENDERBUFFER_HEIGHT: Int\n val RENDERBUFFER_INTERNAL_FORMAT: Int\n val RENDERBUFFER_RED_SIZE: Int\n val RENDERBUFFER_GREEN_SIZE: Int\n val RENDERBUFFER_BLUE_SIZE: Int\n val RENDERBUFFER_ALPHA_SIZE: Int\n val RENDERBUFFER_DEPTH_SIZE: Int\n val RENDERBUFFER_STENCIL_SIZE: Int\n val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int\n val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int\n val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int\n val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int\n val COLOR_ATTACHMENT0: Int\n val DEPTH_ATTACHMENT: Int\n val STENCIL_ATTACHMENT: Int\n val DEPTH_STENCIL_ATTACHMENT: Int\n val NONE: Int\n val FRAMEBUFFER_COMPLETE: Int\n val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int\n val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int\n val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int\n val FRAMEBUFFER_UNSUPPORTED: Int\n val FRAMEBUFFER_BINDING: Int\n val RENDERBUFFER_BINDING: Int\n val MAX_RENDERBUFFER_SIZE: Int\n val INVALID_FRAMEBUFFER_OPERATION: Int\n val UNPACK_FLIP_Y_WEBGL: Int\n val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int\n val CONTEXT_LOST_WEBGL: Int\n val UNPACK_COLORSPACE_CONVERSION_WEBGL: Int\n val BROWSER_DEFAULT_WEBGL: Int\n }\n}\n\n/**\n * Exposes the JavaScript [WebGLContextEvent](https://developer.mozilla.org/en/docs/Web/API/WebGLContextEvent) to Kotlin\n */\npublic external open class WebGLContextEvent(type: String, eventInit: WebGLContextEventInit = definedExternally) : Event {\n open val statusMessage: String\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface WebGLContextEventInit : EventInit {\n var statusMessage: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun WebGLContextEventInit(statusMessage: String? = \"\", bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): WebGLContextEventInit {\n val o = js(\"({})\")\n o[\"statusMessage\"] = statusMessage\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [ArrayBuffer](https://developer.mozilla.org/en/docs/Web/API/ArrayBuffer) to Kotlin\n */\npublic external open class ArrayBuffer(length: Int) : BufferDataSource {\n open val byteLength: Int\n fun slice(begin: Int, end: Int = definedExternally): ArrayBuffer\n\n companion object {\n fun isView(value: Any?): Boolean\n }\n}\n\n/**\n * Exposes the JavaScript [ArrayBufferView](https://developer.mozilla.org/en/docs/Web/API/ArrayBufferView) to Kotlin\n */\npublic external interface ArrayBufferView : BufferDataSource {\n val buffer: ArrayBuffer\n val byteOffset: Int\n val byteLength: Int\n}\n\n/**\n * Exposes the JavaScript [Int8Array](https://developer.mozilla.org/en/docs/Web/API/Int8Array) to Kotlin\n */\npublic external open class Int8Array : ArrayBufferView {\n constructor(length: Int)\n constructor(array: Int8Array)\n constructor(array: Array<Byte>)\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n open val length: Int\n override val buffer: ArrayBuffer\n override val byteOffset: Int\n override val byteLength: Int\n fun set(array: Int8Array, offset: Int = definedExternally)\n fun set(array: Array<Byte>, offset: Int = definedExternally)\n fun subarray(start: Int, end: Int): Int8Array\n\n companion object {\n val BYTES_PER_ELEMENT: Int\n }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",

\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Int8Array.get(index: Int): Byte = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Int8Array.set(index: Int, value: Byte) { asDynamic()[index] = value }\n\n/**\n * Exposes the JavaScript [Uint8Array](https://developer.mozilla.org/en/docs/Web/API/Uint8Array) to Kotlin\n */\npublic external open class Uint8Array : ArrayBufferView {\n   constructor(length: Int)\n   constructor(array: Uint8Array)\n   constructor(array: Array<Byte>)\n   constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n   open val length: Int\n   override val buffer: ArrayBuffer\n   override val byteOffset: Int\n   override val byteLength: Int\n   fun set(array: Uint8Array, offset: Int = definedExternally)\n   fun set(array: Array<Byte>, offset: Int = definedExternally)\n   fun subarray(start: Int, end: Int): Uint8Array\n\n   companion object {\n   val BYTES_PER_ELEMENT: Int\n   }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Uint8Array.get(index: Int): Byte = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Uint8Array.set(index: Int, value: Byte) { asDynamic()[index] = value }\n\n/**\n * Exposes the JavaScript [Uint8ClampedArray](https://developer.mozilla.org/en/docs/Web/API/Uint8ClampedArray) to Kotlin\n */\npublic external open class Uint8ClampedArray : ArrayBufferView {\n   constructor(length: Int)\n   constructor(array: Uint8ClampedArray)\n   constructor(array: Array<Byte>)\n   constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n   open val length: Int\n   override val buffer: ArrayBuffer\n   override val byteOffset: Int\n   override val byteLength: Int\n   fun set(array: Uint8ClampedArray, offset: Int = definedExternally)\n   fun set(array: Array<Byte>, offset: Int = definedExternally)\n   fun subarray(start: Int, end: Int): Uint8ClampedArray\n\n   companion object {\n   val BYTES_PER_ELEMENT: Int\n   }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Uint8ClampedArray.get(index: Int): Byte = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Uint8ClampedArray.set(index: Int, value: Byte) { asDynamic()[index] = value }\n\n/**\n * Exposes the JavaScript [Int16Array](https://developer.mozilla.org/en/docs/Web/API/Int16Array) to Kotlin\n */\npublic external open class Int16Array : ArrayBufferView {\n   constructor(length: Int)\n   constructor(array: Int16Array)\n   constructor(array: Array<Short>)\n   constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n   open val length: Int\n   override val buffer: ArrayBuffer\n   override val byteOffset: Int\n   override val byteLength: Int\n   fun set(array: Int16Array, offset: Int = definedExternally)\n   fun set(array: Array<Short>, offset: Int = definedExternally)\n   fun subarray(start: Int, end: Int): Int16Array\n\n   companion object {\n   val BYTES_PER_ELEMENT: Int\n   }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Int16Array.get(index: Int): Short = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Int16Array.set(index: Int, value: Short) { asDynamic()[index] = value }\n\n/**\n * Exposes the JavaScript [Uint16Array](https://developer.mozilla.org/en/docs/Web/API/Uint16Array) to Kotlin\n */\npublic external open class Uint16Array : ArrayBufferView {\n   constructor(length: Int)\n   constructor(array: Uint16Array)\n   constructor(array: Array<Short>)\n   constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n   open val length: Int\n   override val buffer: ArrayBuffer\n   override val byteOffset: Int\n   override val byteLength: Int\n   fun set(array: Uint16Array, offset: Int = definedExternally)\n   fun set(array: Array<Short>, offset: Int = definedExternally)\n   fun subarray(start: Int, end: Int): Uint16Array\n\n   companion object {\n   val BYTES_PER_ELEMENT: Int\n   }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Uint16Array.get(index: Int): Short = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Uint16Array.set(index: Int,

value: Short) { asDynamic()[index] = value }\n\n/**\n * Exposes the JavaScript [Int32Array](https://developer.mozilla.org/en/docs/Web/API/Int32Array) to Kotlin\n */\npublic external open class Int32Array : ArrayBufferView {\n constructor(length: Int)\n constructor(array: Int32Array)\n constructor(array: Array<Int>)\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n open val length: Int\n override val buffer: ArrayBuffer\n override val byteOffset: Int\n override val byteLength: Int\n fun set(array: Int32Array, offset: Int = definedExternally)\n fun set(array: Array<Int>, offset: Int = definedExternally)\n fun subarray(start: Int, end: Int): Int32Array\n\n companion object {\n val BYTES_PER_ELEMENT: Int\n }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Int32Array.get(index: Int): Int = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Int32Array.set(index: Int, value: Int) { asDynamic()[index] = value }\n\n/**\n * Exposes the JavaScript [Uint32Array](https://developer.mozilla.org/en/docs/Web/API/Uint32Array) to Kotlin\n */\npublic external open class Uint32Array : ArrayBufferView {\n constructor(length: Int)\n constructor(array: Uint32Array)\n constructor(array: Array<Int>)\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n open val length: Int\n override val buffer: ArrayBuffer\n override val byteOffset: Int\n override val byteLength: Int\n fun set(array: Uint32Array, offset: Int = definedExternally)\n fun set(array: Array<Int>, offset: Int = definedExternally)\n fun subarray(start: Int, end: Int): Uint32Array\n\n companion object {\n val BYTES_PER_ELEMENT: Int\n }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Uint32Array.get(index: Int): Int = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Uint32Array.set(index: Int, value: Int) { asDynamic()[index] = value }\n\n/**\n * Exposes the JavaScript [Float32Array](https://developer.mozilla.org/en/docs/Web/API/Float32Array) to Kotlin\n */\npublic external open class Float32Array : ArrayBufferView {\n constructor(length: Int)\n constructor(array: Float32Array)\n constructor(array: Array<Float>)\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n open val length: Int\n override val buffer: ArrayBuffer\n override val byteOffset: Int\n override val byteLength: Int\n fun set(array: Float32Array, offset: Int = definedExternally)\n fun set(array: Array<Float>, offset: Int = definedExternally)\n fun subarray(start: Int, end: Int): Float32Array\n\n companion object {\n val BYTES_PER_ELEMENT: Int\n }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Float32Array.get(index: Int): Float = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Float32Array.set(index: Int, value: Float) { asDynamic()[index] = value }\n\n/**\n * Exposes the JavaScript [Float64Array](https://developer.mozilla.org/en/docs/Web/API/Float64Array) to Kotlin\n */\npublic external open class Float64Array : ArrayBufferView {\n constructor(length: Int)\n constructor(array: Float64Array)\n constructor(array: Array<Double>)\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n open val length: Int\n override val buffer: ArrayBuffer\n override val byteOffset: Int\n override val byteLength: Int\n fun set(array: Float64Array, offset: Int = definedExternally)\n fun set(array: Array<Double>, offset: Int = definedExternally)\n fun subarray(start: Int, end: Int): Float64Array\n\n companion object {\n val BYTES_PER_ELEMENT: Int\n }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Float64Array.get(index: Int): Double = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Float64Array.set(index: Int, value: Double) { asDynamic()[index] = value }\n\n/**\n * Exposes the JavaScript [DataView](https://developer.mozilla.org/en/docs/Web/API/DataView) to Kotlin\n */\npublic external open class

DataView(buffer: ArrayBuffer, byteOffset: Int = definedExternally, byteLength: Int = definedExternally) : ArrayBufferView {\n   override val buffer: ArrayBuffer\n   override val byteOffset: Int\n   override val byteLength: Int\n   fun getInt8(byteOffset: Int): Byte\n   fun getUint8(byteOffset: Int): Byte\n   fun getInt16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short\n   fun getUint16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short\n   fun getInt32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int\n   fun getUint32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int\n   fun getFloat32(byteOffset: Int, littleEndian: Boolean = definedExternally): Float\n   fun getFloat64(byteOffset: Int, littleEndian: Boolean = definedExternally): Double\n   fun setInt8(byteOffset: Int, value: Byte)\n   fun setUint8(byteOffset: Int, value: Byte)\n   fun setInt16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally)\n   fun setUint16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally)\n   fun setInt32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)\n   fun setUint32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)\n   fun setFloat32(byteOffset: Int, value: Float, littleEndian: Boolean = definedExternally)\n   fun setFloat64(byteOffset: Int, value: Double, littleEndian: Boolean = definedExternally)\n}\n\npublic external interface BufferDataSource\n\npublic external interface TexImageSource","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.dom.clipboard\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\nimport org.w3c.dom.events.*\n\npublic external interface ClipboardEventInit : EventInit {\n   var clipboardData: DataTransfer? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ClipboardEventInit(clipboardData: DataTransfer? = null, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): ClipboardEventInit {\n   val o = js(\"({})\")\n   o[\"clipboardData\"] = clipboardData\n   o[\"bubbles\"] = bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\n/**\n * Exposes the JavaScript [ClipboardEvent](https://developer.mozilla.org/en/docs/Web/API/ClipboardEvent) to Kotlin\n */\npublic external open class ClipboardEvent(type: String, eventInitDict: ClipboardEventInit = definedExternally) : Event {\n   open val clipboardData: DataTransfer?\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [Clipboard](https://developer.mozilla.org/en/docs/Web/API/Clipboard) to Kotlin\n */\npublic external abstract class Clipboard : EventTarget {\n   fun read(): Promise<DataTransfer>\n   fun readText(): Promise<String>\n   fun write(data: DataTransfer): Promise<Unit>\n   fun writeText(data: String): Promise<Unit>\n}\n\npublic external interface ClipboardPermissionDescriptor {\n   var allowWithoutGesture: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ClipboardPermissionDescriptor(allowWithoutGesture: Boolean? = false): ClipboardPermissionDescriptor {\n   val o = js(\"({})\")\n   o[\"allowWithoutGesture\"] = allowWithoutGesture\n   return o\n}","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.dom.css\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\n\npublic external abstract class MediaList : ItemArrayLike<String> {\n   open var mediaText: String\n   fun appendMedium(medium: String)\n   fun deleteMedium(medium: String)\n   override fun item(index: Int): String?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun MediaList.get(index: Int): String? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript [StyleSheet](https://developer.mozilla.org/en/docs/Web/API/StyleSheet) to Kotlin\n */\npublic external abstract

```
class StyleSheet {\n    open val type: String\n    open val href: String?\n    open val ownerNode:
UnionElementOrProcessingInstruction?\n    open val parentStyleSheet: StyleSheet?\n    open val title: String?\n
open val media: MediaList\n    open var disabled: Boolean\n}\n\n/**\n * Exposes the JavaScript
[CSSStyleSheet](https://developer.mozilla.org/en/docs/Web/API/CSSStyleSheet) to Kotlin\n */\npublic external
abstract class CSSStyleSheet : StyleSheet {\n    open val ownerRule: CSSRule?\n    open val cssRules:
CSSRuleList\n    fun insertRule(rule: String, index: Int): Int\n    fun deleteRule(index: Int)\n}\n\n/**\n * Exposes the
JavaScript [StyleSheetList](https://developer.mozilla.org/en/docs/Web/API/StyleSheetList) to Kotlin\n */\npublic
external abstract class StyleSheetList : ItemArrayLike<StyleSheet> {\n    override fun item(index: Int):
StyleSheet?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun StyleSheetList.get(index: Int):
StyleSheet? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript
[LinkStyle](https://developer.mozilla.org/en/docs/Web/API/LinkStyle) to Kotlin\n */\npublic external interface
LinkStyle {\n    val sheet: StyleSheet?\n        get() = definedExternally\n}\n\n/**\n * Exposes the JavaScript
[CSSRuleList](https://developer.mozilla.org/en/docs/Web/API/CSSRuleList) to Kotlin\n */\npublic external abstract
class CSSRuleList : ItemArrayLike<CSSRule> {\n    override fun item(index: Int):
CSSRule?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun CSSRuleList.get(index: Int):
CSSRule? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript
[CSSRule](https://developer.mozilla.org/en/docs/Web/API/CSSRule) to Kotlin\n */\npublic external abstract class
CSSRule {\n    open val type: Short\n    open var cssText: String\n    open val parentRule: CSSRule?\n    open val
parentStyleSheet: CSSStyleSheet?\n\n    companion object {\n        val STYLE_RULE: Short\n        val
CHARSET_RULE: Short\n        val IMPORT_RULE: Short\n        val MEDIA_RULE: Short\n        val
FONT_FACE_RULE: Short\n        val PAGE_RULE: Short\n        val MARGIN_RULE: Short\n        val
NAMESPACE_RULE: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[CSSStyleRule](https://developer.mozilla.org/en/docs/Web/API/CSSStyleRule) to Kotlin\n */\npublic external
abstract class CSSStyleRule : CSSRule {\n    open var selectorText: String\n    open val style:
CSSStyleDeclaration\n\n    companion object {\n        val STYLE_RULE: Short\n        val CHARSET_RULE:
Short\n        val IMPORT_RULE: Short\n        val MEDIA_RULE: Short\n        val FONT_FACE_RULE: Short\n
  val PAGE_RULE: Short\n        val MARGIN_RULE: Short\n        val NAMESPACE_RULE: Short\n
}\n}\n\npublic external abstract class CSSImportRule : CSSRule {\n    open val href: String\n    open val media:
MediaList\n    open val styleSheet: CSSStyleSheet\n\n    companion object {\n        val STYLE_RULE: Short\n
val CHARSET_RULE: Short\n        val IMPORT_RULE: Short\n        val MEDIA_RULE: Short\n        val
FONT_FACE_RULE: Short\n        val PAGE_RULE: Short\n        val MARGIN_RULE: Short\n        val
NAMESPACE_RULE: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[CSSGroupingRule](https://developer.mozilla.org/en/docs/Web/API/CSSGroupingRule) to Kotlin\n */\npublic
external abstract class CSSGroupingRule : CSSRule {\n    open val cssRules: CSSRuleList\n    fun insertRule(rule:
String, index: Int): Int\n    fun deleteRule(index: Int)\n\n    companion object {\n        val STYLE_RULE: Short\n
 val CHARSET_RULE: Short\n        val IMPORT_RULE: Short\n        val MEDIA_RULE: Short\n        val
FONT_FACE_RULE: Short\n        val PAGE_RULE: Short\n        val MARGIN_RULE: Short\n        val
NAMESPACE_RULE: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[CSSMediaRule](https://developer.mozilla.org/en/docs/Web/API/CSSMediaRule) to Kotlin\n */\npublic external
abstract class CSSMediaRule : CSSGroupingRule {\n    open val media: MediaList\n\n    companion object {\n
val STYLE_RULE: Short\n        val CHARSET_RULE: Short\n        val IMPORT_RULE: Short\n        val
MEDIA_RULE: Short\n        val FONT_FACE_RULE: Short\n        val PAGE_RULE: Short\n        val
MARGIN_RULE: Short\n        val NAMESPACE_RULE: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[CSSPageRule](https://developer.mozilla.org/en/docs/Web/API/CSSPageRule) to Kotlin\n */\npublic external
abstract class CSSPageRule : CSSGroupingRule {\n    open var selectorText: String\n    open val style:
CSSStyleDeclaration\n\n    companion object {\n        val STYLE_RULE: Short\n        val CHARSET_RULE:
```

Short\n val IMPORT_RULE: Short\n val MEDIA_RULE: Short\n val FONT_FACE_RULE: Short\n val PAGE_RULE: Short\n val MARGIN_RULE: Short\n val NAMESPACE_RULE: Short\n }\n}\n\npublic external abstract class CSSMarginRule : CSSRule {\n open val name: String\n open val style: CSSStyleDeclaration\n\n companion object {\n val STYLE_RULE: Short\n val CHARSET_RULE: Short\n val IMPORT_RULE: Short\n val MEDIA_RULE: Short\n val FONT_FACE_RULE: Short\n val PAGE_RULE: Short\n val MARGIN_RULE: Short\n val NAMESPACE_RULE: Short\n }\n}\n\n/**\n * Exposes the JavaScript [CSSNamespaceRule](https://developer.mozilla.org/en/docs/Web/API/CSSNamespaceRule) to Kotlin\n */\npublic external abstract class CSSNamespaceRule : CSSRule {\n open val namespaceURI: String\n open val prefix: String\n\n companion object {\n val STYLE_RULE: Short\n val CHARSET_RULE: Short\n val IMPORT_RULE: Short\n val MEDIA_RULE: Short\n val FONT_FACE_RULE: Short\n val PAGE_RULE: Short\n val MARGIN_RULE: Short\n val NAMESPACE_RULE: Short\n }\n}\n\n/**\n * Exposes the JavaScript [CSSStyleDeclaration](https://developer.mozilla.org/en/docs/Web/API/CSSStyleDeclaration) to Kotlin\n */\npublic external abstract class CSSStyleDeclaration : ItemArrayLike<String> {\n open var cssText: String\n open val parentRule: CSSRule?\n open var cssFloat: String\n open var alignContent: String\n open var alignItems: String\n open var alignSelf: String\n open var animation: String\n open var animationDelay: String\n open var animationDirection: String\n open var animationDuration: String\n open var animationFillMode: String\n open var animationIterationCount: String\n open var animationName: String\n open var animationPlayState: String\n open var animationTimingFunction: String\n open var backfaceVisibility: String\n open var background: String\n open var backgroundAttachment: String\n open var backgroundClip: String\n open var backgroundColor: String\n open var backgroundImage: String\n open var backgroundOrigin: String\n open var backgroundPosition: String\n open var backgroundRepeat: String\n open var backgroundSize: String\n open var border: String\n open var borderBottom: String\n open var borderBottomColor: String\n open var borderBottomLeftRadius: String\n open var borderBottomRightRadius: String\n open var borderBottomStyle: String\n open var borderBottomWidth: String\n open var borderCollapse: String\n open var borderColor: String\n open var borderImage: String\n open var borderImageOutset: String\n open var borderImageRepeat: String\n open var borderImageSlice: String\n open var borderImageSource: String\n open var borderImageWidth: String\n open var borderLeft: String\n open var borderLeftColor: String\n open var borderLeftStyle: String\n open var borderLeftWidth: String\n open var borderRadius: String\n open var borderRight: String\n open var borderRightColor: String\n open var borderRightStyle: String\n open var borderRightWidth: String\n open var borderSpacing: String\n open var borderStyle: String\n open var borderTop: String\n open var borderTopColor: String\n open var borderTopLeftRadius: String\n open var borderTopRightRadius: String\n open var borderTopStyle: String\n open var borderTopWidth: String\n open var borderWidth: String\n open var bottom: String\n open var boxDecorationBreak: String\n open var boxShadow: String\n open var boxSizing: String\n open var breakAfter: String\n open var breakBefore: String\n open var breakInside: String\n open var captionSide: String\n open var clear: String\n open var clip: String\n open var color: String\n open var columnCount: String\n open var columnFill: String\n open var columnGap: String\n open var columnRule: String\n open var columnRuleColor: String\n open var columnRuleStyle: String\n open var columnRuleWidth: String\n open var columnSpan: String\n open var columnWidth: String\n open var columns: String\n open var content: String\n open var counterIncrement: String\n open var counterReset: String\n open var cursor: String\n open var direction: String\n open var display: String\n open var emptyCells: String\n open var filter: String\n open var flex: String\n open var flexBasis: String\n open var flexDirection: String\n open var flexFlow: String\n open var flexGrow: String\n open var flexShrink: String\n open var flexWrap: String\n open var font: String\n open var fontFamily: String\n open var fontFeatureSettings: String\n open var fontKerning: String\n open var fontLanguageOverride: String\n open var fontSize: String\n open var fontSizeAdjust: String\n open var fontStretch: String\n open var fontStyle: String\n open var fontSynthesis: String\n open var fontVariant:

String\n   open var fontVariantAlternates: String\n   open var fontVariantCaps: String\n   open var fontVariantEastAsian: String\n   open var fontVariantLigatures: String\n   open var fontVariantNumeric: String\n   open var fontVariantPosition: String\n   open var fontWeight: String\n   open var hangingPunctuation: String\n   open var height: String\n   open var hyphens: String\n   open var imageOrientation: String\n   open var imageRendering: String\n   open var imageResolution: String\n   open var imeMode: String\n   open var justifyContent: String\n   open var left: String\n   open var letterSpacing: String\n   open var lineBreak: String\n   open var lineHeight: String\n   open var listStyle: String\n   open var listStyleImage: String\n   open var listStylePosition: String\n   open var listStyleType: String\n   open var margin: String\n   open var marginBottom: String\n   open var marginLeft: String\n   open var marginRight: String\n   open var marginTop: String\n   open var mark: String\n   open var markAfter: String\n   open var markBefore: String\n   open var marks: String\n   open var marqueeDirection: String\n   open var marqueePlayCount: String\n   open var marqueeSpeed: String\n   open var marqueeStyle: String\n   open var mask: String\n   open var maskType: String\n   open var maxHeight: String\n   open var maxWidth: String\n   open var minHeight: String\n   open var minWidth: String\n   open var navDown: String\n   open var navIndex: String\n   open var navLeft: String\n   open var navRight: String\n   open var navUp: String\n   open var objectFit: String\n   open var objectPosition: String\n   open var opacity: String\n   open var order: String\n   open var orphans: String\n   open var outline: String\n   open var outlineColor: String\n   open var outlineOffset: String\n   open var outlineStyle: String\n   open var outlineWidth: String\n   open var overflowWrap: String\n   open var overflowX: String\n   open var overflowY: String\n   open var padding: String\n   open var paddingBottom: String\n   open var paddingLeft: String\n   open var paddingRight: String\n   open var paddingTop: String\n   open var pageBreakAfter: String\n   open var pageBreakBefore: String\n   open var pageBreakInside: String\n   open var perspective: String\n   open var perspectiveOrigin: String\n   open var phonemes: String\n   open var position: String\n   open var quotes: String\n   open var resize: String\n   open var rest: String\n   open var restAfter: String\n   open var restBefore: String\n   open var right: String\n   open var tabSize: String\n   open var tableLayout: String\n   open var textAlign: String\n   open var textAlignLast: String\n   open var textCombineUpright: String\n   open var textDecoration: String\n   open var textDecorationColor: String\n   open var textDecorationLine: String\n   open var textDecorationStyle: String\n   open var textIndent: String\n   open var textJustify: String\n   open var textOrientation: String\n   open var textOverflow: String\n   open var textShadow: String\n   open var textTransform: String\n   open var textUnderlinePosition: String\n   open var top: String\n   open var transform: String\n   open var transformOrigin: String\n   open var transformStyle: String\n   open var transition: String\n   open var transitionDelay: String\n   open var transitionDuration: String\n   open var transitionProperty: String\n   open var transitionTimingFunction: String\n   open var unicodeBidi: String\n   open var verticalAlign: String\n   open var visibility: String\n   open var voiceBalance: String\n   open var voiceDuration: String\n   open var voicePitch: String\n   open var voicePitchRange: String\n   open var voiceRate: String\n   open var voiceStress: String\n   open var voiceVolume: String\n   open var whiteSpace: String\n   open var widows: String\n   open var width: String\n   open var wordBreak: String\n   open var wordSpacing: String\n   open var wordWrap: String\n   open var writingMode: String\n   open var zIndex: String\n   open var _dashed_attribute: String\n   open var _camel_cased_attribute: String\n   open var _webkit_cased_attribute: String\n   fun getPropertyValue(property: String): String\n   fun getPropertyPriority(property: String): String\n   fun setProperty(property: String, value: String, priority: String = definedExternally)\n   fun setPropertyValue(property: String, value: String)\n   fun setPropertyPriority(property: String, priority: String)\n   fun removeProperty(property: String): String\n   override fun item(index: Int): String\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun CSSStyleDeclaration.get(index: Int): String? = asDynamic()[index]\n\npublic external interface ElementCSSInlineStyle {\n   val style: CSSStyleDeclaration\n}\n\n/**\n * Exposes the JavaScript [CSS](https://developer.mozilla.org/en/docs/Web/API/CSS) to Kotlin\n */\npublic external abstract class CSS {\n   companion object {\n   fun escape(ident: String): String\n   }\n}\n\npublic external interface UnionElementOrProcessingInstruction","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming

Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.dom.encryptedmedia\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\nimport org.w3c.dom.events.*\n\n/**\n * Exposes the JavaScript [MediaKeySystemConfiguration](https://developer.mozilla.org/en/docs/Web/API/MediaKeySystemConfiguration) to Kotlin\n */\npublic external interface MediaKeySystemConfiguration {\n var label: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n var initDataTypes: Array<String>? /* = arrayOf() */\n get() = definedExternally\n set(value) = definedExternally\n var audioCapabilities: Array<MediaKeySystemMediaCapability>? /* = arrayOf() */\n get() = definedExternally\n set(value) = definedExternally\n var videoCapabilities: Array<MediaKeySystemMediaCapability>? /* = arrayOf() */\n get() = definedExternally\n set(value) = definedExternally\n var distinctiveIdentifier: MediaKeysRequirement? /* = MediaKeysRequirement.OPTIONAL */\n get() = definedExternally\n set(value) = definedExternally\n var persistentState: MediaKeysRequirement? /* = MediaKeysRequirement.OPTIONAL */\n get() = definedExternally\n set(value) = definedExternally\n var sessionTypes: Array<String>?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaKeySystemConfiguration(label: String? = \"\", initDataTypes: Array<String>? = arrayOf(), audioCapabilities: Array<MediaKeySystemMediaCapability>? = arrayOf(), videoCapabilities: Array<MediaKeySystemMediaCapability>? = arrayOf(), distinctiveIdentifier: MediaKeysRequirement? = MediaKeysRequirement.OPTIONAL, persistentState: MediaKeysRequirement? = MediaKeysRequirement.OPTIONAL, sessionTypes: Array<String>? = undefined): MediaKeySystemConfiguration {\n val o = js(\"({})\")\n o[\"label\"] = label\n o[\"initDataTypes\"] = initDataTypes\n o[\"audioCapabilities\"] = audioCapabilities\n o[\"videoCapabilities\"] = videoCapabilities\n o[\"distinctiveIdentifier\"] = distinctiveIdentifier\n o[\"persistentState\"] = persistentState\n o[\"sessionTypes\"] = sessionTypes\n return o\n}\n\npublic external interface MediaKeySystemMediaCapability {\n var contentType: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n var robustness: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaKeySystemMediaCapability(contentType: String? = \"\", robustness: String? = \"\"): MediaKeySystemMediaCapability {\n val o = js(\"({})\")\n o[\"contentType\"] = contentType\n o[\"robustness\"] = robustness\n return o\n}\n\n/**\n * Exposes the JavaScript [MediaKeySystemAccess](https://developer.mozilla.org/en/docs/Web/API/MediaKeySystemAccess) to Kotlin\n */\npublic external abstract class MediaKeySystemAccess {\n open val keySystem: String\n fun getConfiguration(): MediaKeySystemConfiguration\n fun createMediaKeys(): Promise<MediaKeys>\n}\n\n/**\n * Exposes the JavaScript [MediaKeys](https://developer.mozilla.org/en/docs/Web/API/MediaKeys) to Kotlin\n */\npublic external abstract class MediaKeys {\n fun createSession(sessionType: MediaKeySessionType = definedExternally): MediaKeySession\n fun setServerCertificate(serverCertificate: dynamic): Promise<Boolean>\n}\n\n/**\n * Exposes the JavaScript [MediaKeySession](https://developer.mozilla.org/en/docs/Web/API/MediaKeySession) to Kotlin\n */\npublic external abstract class MediaKeySession : EventTarget {\n open val sessionId: String\n open val expiration: Double\n open val closed: Promise<Unit>\n open val keyStatuses: MediaKeyStatusMap\n open var onkeystatuseschange: ((Event) -> dynamic)?\n open var onmessage: ((MessageEvent) -> dynamic)?\n fun generateRequest(initDataType: String, initData: dynamic): Promise<Unit>\n fun load(sessionId: String): Promise<Boolean>\n fun update(response: dynamic): Promise<Unit>\n fun close(): Promise<Unit>\n fun remove(): Promise<Unit>\n}\n\n/**\n * Exposes the JavaScript [MediaKeyStatusMap](https://developer.mozilla.org/en/docs/Web/API/MediaKeyStatusMap) to Kotlin\n */\npublic

external abstract class MediaKeyStatusMap {\n   open val size: Int\n   fun has(keyId: dynamic): Boolean\n   fun get(keyId: dynamic): Any?\n}\n\n/**\n * Exposes the JavaScript [MediaKeyMessageEvent](https://developer.mozilla.org/en/docs/Web/API/MediaKeyMessageEvent) to Kotlin\n */\npublic external open class MediaKeyMessageEvent(type: String, eventInitDict: MediaKeyMessageEventInit) : Event {\n   open val messageType: MediaKeyMessageType\n   open val message: ArrayBuffer\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface MediaKeyMessageEventInit : EventInit {\n   var messageType: MediaKeyMessageType?\n   var message: ArrayBuffer?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaKeyMessageEventInit(messageType: MediaKeyMessageType?, message: ArrayBuffer?, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): MediaKeyMessageEventInit {\n   val o = js(\"({})\")\n   o[\"messageType\"] = messageType\n   o[\"message\"] = message\n   o[\"bubbles\"] = bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\npublic external open class MediaEncryptedEvent(type: String, eventInitDict: MediaEncryptedEventInit = definedExternally) : Event {\n   open val initDataType: String\n   open val initData: ArrayBuffer?\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface MediaEncryptedEventInit : EventInit {\n   var initDataType: String? /* = \"\" */\n   get() = definedExternally\n   set(value) = definedExternally\n   var initData: ArrayBuffer? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaEncryptedEventInit(initDataType: String? = \"\", initData: ArrayBuffer? = null, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): MediaEncryptedEventInit {\n   val o = js(\"({})\")\n   o[\"initDataType\"] = initDataType\n   o[\"initData\"] = initData\n   o[\"bubbles\"] = bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface MediaKeysRequirement {\n   companion object\n}\n\npublic inline val MediaKeysRequirement.Companion.REQUIRED: MediaKeysRequirement get() = \"required\".asDynamic().unsafeCast<MediaKeysRequirement>()\n\npublic inline val MediaKeysRequirement.Companion.OPTIONAL: MediaKeysRequirement get() = \"optional\".asDynamic().unsafeCast<MediaKeysRequirement>()\n\npublic inline val MediaKeysRequirement.Companion.NOT_ALLOWED: MediaKeysRequirement get() = \"not-allowed\".asDynamic().unsafeCast<MediaKeysRequirement>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface MediaKeySessionType {\n   companion object\n}\n\npublic inline val MediaKeySessionType.Companion.TEMPORARY: MediaKeySessionType get() = \"temporary\".asDynamic().unsafeCast<MediaKeySessionType>()\n\npublic inline val MediaKeySessionType.Companion.PERSISTENT_LICENSE: MediaKeySessionType get() = \"persistent-license\".asDynamic().unsafeCast<MediaKeySessionType>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface MediaKeyStatus {\n   companion object\n}\n\npublic inline val MediaKeyStatus.Companion.USABLE: MediaKeyStatus get() = \"usable\".asDynamic().unsafeCast<MediaKeyStatus>()\n\npublic inline val MediaKeyStatus.Companion.EXPIRED: MediaKeyStatus get() = \"expired\".asDynamic().unsafeCast<MediaKeyStatus>()\n\npublic inline val MediaKeyStatus.Companion.RELEASED: MediaKeyStatus get() = \"released\".asDynamic().unsafeCast<MediaKeyStatus>()\n\npublic inline val MediaKeyStatus.Companion.OUTPUT_RESTRICTED: MediaKeyStatus get() = \"output-

restricted\".asDynamic().unsafeCast<MediaKeyStatus>()\n\npublic inline val MediaKeyStatus.Companion.OUTPUT_DOWNSCALED: MediaKeyStatus get() = \"output-downscaled\".asDynamic().unsafeCast<MediaKeyStatus>()\n\npublic inline val MediaKeyStatus.Companion.STATUS_PENDING: MediaKeyStatus get() = \"status-pending\".asDynamic().unsafeCast<MediaKeyStatus>()\n\npublic inline val MediaKeyStatus.Companion.INTERNAL_ERROR: MediaKeyStatus get() = \"internal-error\".asDynamic().unsafeCast<MediaKeyStatus>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface MediaKeyMessageType {\n   companion object\n}\n\npublic inline val MediaKeyMessageType.Companion.LICENSE_REQUEST: MediaKeyMessageType get() = \"license-request\".asDynamic().unsafeCast<MediaKeyMessageType>()\n\npublic inline val MediaKeyMessageType.Companion.LICENSE_RENEWAL: MediaKeyMessageType get() = \"license-renewal\".asDynamic().unsafeCast<MediaKeyMessageType>()\n\npublic inline val MediaKeyMessageType.Companion.LICENSE_RELEASE: MediaKeyMessageType get() = \"license-release\".asDynamic().unsafeCast<MediaKeyMessageType>()\n\npublic inline val MediaKeyMessageType.Companion.INDIVIDUALIZATION_REQUEST: MediaKeyMessageType get() = \"individualization-request\".asDynamic().unsafeCast<MediaKeyMessageType>()","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.dom.events\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\n\n/**\n * Exposes the JavaScript [UIEvent](https://developer.mozilla.org/en/docs/Web/API/UIEvent) to Kotlin\n */\npublic external open class UIEvent(type: String, eventInitDict: UIEventInit = definedExternally) : Event {\n   open val view: Window?\n   open val detail: Int\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface UIEventInit : EventInit {\n   var view: Window? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n   var detail: Int? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun UIEventInit(view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): UIEventInit {\n   val o = js(\"({})\")\n   o[\"view\"] = view\n   o[\"detail\"] = detail\n   o[\"bubbles\"] = bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\n/**\n * Exposes the JavaScript [FocusEvent](https://developer.mozilla.org/en/docs/Web/API/FocusEvent) to Kotlin\n */\npublic external open class FocusEvent(type: String, eventInitDict: FocusEventInit = definedExternally) : UIEvent {\n   open val relatedTarget: EventTarget?\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface FocusEventInit : UIEventInit {\n   var relatedTarget: EventTarget? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun FocusEventInit(relatedTarget: EventTarget? = null, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): FocusEventInit {\n   val o = js(\"({})\")\n   o[\"relatedTarget\"] = relatedTarget\n   o[\"view\"] = view\n   o[\"detail\"] = detail\n   o[\"bubbles\"] = bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\n/**\n * Exposes the JavaScript [MouseEvent](https://developer.mozilla.org/en/docs/Web/API/MouseEvent) to Kotlin\n */\npublic external open class MouseEvent(type: String, eventInitDict: MouseEventInit = definedExternally) : UIEvent, UnionElementOrMouseEvent {\n   open val screenX: Int\n   open val screenY: Int\n   open val clientX: Int\n   open val clientY: Int\n   open val ctrlKey: Boolean\n   open val shiftKey: Boolean\n   open val altKey: Boolean\n   open val metaKey: Boolean\n   open val button: Short\n   open val buttons: Short\n   open val relatedTarget:

EventTarget?\n   open val region: String?\n   open val pageX: Double\n   open val pageY: Double\n   open val x: Double\n   open val y: Double\n   open val offsetX: Double\n   open val offsetY: Double\n   fun getModifierState(keyArg: String): Boolean\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface MouseEventInit : EventModifierInit {\n   var screenX: Int? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var screenY: Int? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var clientX: Int? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var clientY: Int? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var button: Short? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var buttons: Short? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var relatedTarget: EventTarget? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n   var region: String? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MouseEventInit(screenX: Int? = 0, screenY: Int? = 0, clientX: Int? = 0, clientY: Int? = 0, button: Short? = 0, buttons: Short? = 0, relatedTarget: EventTarget? = null, region: String? = null, ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false, metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn: Boolean? = false, modifierFnLock: Boolean? = false, modifierHyper: Boolean? = false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false, modifierSymbol: Boolean? = false, modifierSymbolLock: Boolean? = false, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): MouseEventInit {\n   val o = js(\"({})\")\n   o[\"screenX\"] = screenX\n   o[\"screenY\"] = screenY\n   o[\"clientX\"] = clientX\n   o[\"clientY\"] = clientY\n   o[\"button\"] = button\n   o[\"buttons\"] = buttons\n   o[\"relatedTarget\"] = relatedTarget\n   o[\"region\"] = region\n   o[\"ctrlKey\"] = ctrlKey\n   o[\"shiftKey\"] = shiftKey\n   o[\"altKey\"] = altKey\n   o[\"metaKey\"] = metaKey\n   o[\"modifierAltGraph\"] = modifierAltGraph\n   o[\"modifierCapsLock\"] = modifierCapsLock\n   o[\"modifierFn\"] = modifierFn\n   o[\"modifierFnLock\"] = modifierFnLock\n   o[\"modifierHyper\"] = modifierHyper\n   o[\"modifierNumLock\"] = modifierNumLock\n   o[\"modifierScrollLock\"] = modifierScrollLock\n   o[\"modifierSuper\"] = modifierSuper\n   o[\"modifierSymbol\"] = modifierSymbol\n   o[\"modifierSymbolLock\"] = modifierSymbolLock\n   o[\"view\"] = view\n   o[\"detail\"] = detail\n   o[\"bubbles\"] = bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\npublic external interface EventModifierInit : UIEventInit {\n   var ctrlKey: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var shiftKey: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var altKey: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var metaKey: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var modifierAltGraph: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var modifierCapsLock: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var modifierFn: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var modifierFnLock: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var modifierHyper: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var modifierNumLock: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var modifierScrollLock: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var modifierSuper: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var modifierSymbol: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var modifierSymbolLock: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun EventModifierInit(ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false, metaKey: Boolean? = false, modifierAltGraph:

```
Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn: Boolean? = false, modifierFnLock: Boolean? = false, modifierHyper: Boolean? = false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false, modifierSymbol: Boolean? = false, modifierSymbolLock: Boolean? = false, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): EventModifierInit {\n    val o = js(\"({})\")\n    o[\"ctrlKey\"] = ctrlKey\n    o[\"shiftKey\"] = shiftKey\n    o[\"altKey\"] = altKey\n    o[\"metaKey\"] = metaKey\n    o[\"modifierAltGraph\"] = modifierAltGraph\n    o[\"modifierCapsLock\"] = modifierCapsLock\n    o[\"modifierFn\"] = modifierFn\n    o[\"modifierFnLock\"] = modifierFnLock\n    o[\"modifierHyper\"] = modifierHyper\n    o[\"modifierNumLock\"] = modifierNumLock\n    o[\"modifierScrollLock\"] = modifierScrollLock\n    o[\"modifierSuper\"] = modifierSuper\n    o[\"modifierSymbol\"] = modifierSymbol\n    o[\"modifierSymbolLock\"] = modifierSymbolLock\n    o[\"view\"] = view\n    o[\"detail\"] = detail\n    o[\"bubbles\"] = bubbles\n    o[\"cancelable\"] = cancelable\n    o[\"composed\"] = composed\n    return o\n}\n\n/**\n * Exposes the JavaScript [WheelEvent](https://developer.mozilla.org/en/docs/Web/API/WheelEvent) to Kotlin\n */\npublic external open class WheelEvent(type: String, eventInitDict: WheelEventInit = definedExternally) : MouseEvent {\n    open val deltaX: Double\n    open val deltaY: Double\n    open val deltaZ: Double\n    open val deltaMode: Int\n\n    companion object {\n        val DOM_DELTA_PIXEL: Int\n        val DOM_DELTA_LINE: Int\n        val DOM_DELTA_PAGE: Int\n        val NONE: Short\n        val CAPTURING_PHASE: Short\n        val AT_TARGET: Short\n        val BUBBLING_PHASE: Short\n    }\n}\n\npublic external interface WheelEventInit : MouseEventInit {\n    var deltaX: Double? /* = 0.0 */\n        get() = definedExternally\n        set(value) = definedExternally\n    var deltaY: Double? /* = 0.0 */\n        get() = definedExternally\n        set(value) = definedExternally\n    var deltaZ: Double? /* = 0.0 */\n        get() = definedExternally\n        set(value) = definedExternally\n    var deltaMode: Int? /* = 0 */\n        get() = definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun WheelEventInit(deltaX: Double? = 0.0, deltaY: Double? = 0.0, deltaZ: Double? = 0.0, deltaMode: Int? = 0, screenX: Int? = 0, screenY: Int? = 0, clientX: Int? = 0, clientY: Int? = 0, button: Short? = 0, buttons: Short? = 0, relatedTarget: EventTarget? = null, region: String? = null, ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false, metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn: Boolean? = false, modifierFnLock: Boolean? = false, modifierHyper: Boolean? = false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false, modifierSymbol: Boolean? = false, modifierSymbolLock: Boolean? = false, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): WheelEventInit {\n    val o = js(\"({})\")\n    o[\"deltaX\"] = deltaX\n    o[\"deltaY\"] = deltaY\n    o[\"deltaZ\"] = deltaZ\n    o[\"deltaMode\"] = deltaMode\n    o[\"screenX\"] = screenX\n    o[\"screenY\"] = screenY\n    o[\"clientX\"] = clientX\n    o[\"clientY\"] = clientY\n    o[\"button\"] = button\n    o[\"buttons\"] = buttons\n    o[\"relatedTarget\"] = relatedTarget\n    o[\"region\"] = region\n    o[\"ctrlKey\"] = ctrlKey\n    o[\"shiftKey\"] = shiftKey\n    o[\"altKey\"] = altKey\n    o[\"metaKey\"] = metaKey\n    o[\"modifierAltGraph\"] = modifierAltGraph\n    o[\"modifierCapsLock\"] = modifierCapsLock\n    o[\"modifierFn\"] = modifierFn\n    o[\"modifierFnLock\"] = modifierFnLock\n    o[\"modifierHyper\"] = modifierHyper\n    o[\"modifierNumLock\"] = modifierNumLock\n    o[\"modifierScrollLock\"] = modifierScrollLock\n    o[\"modifierSuper\"] = modifierSuper\n    o[\"modifierSymbol\"] = modifierSymbol\n    o[\"modifierSymbolLock\"] = modifierSymbolLock\n    o[\"view\"] = view\n    o[\"detail\"] = detail\n    o[\"bubbles\"] = bubbles\n    o[\"cancelable\"] = cancelable\n    o[\"composed\"] = composed\n    return o\n}\n\n/**\n * Exposes the JavaScript [InputEvent](https://developer.mozilla.org/en/docs/Web/API/InputEvent) to Kotlin\n */\npublic external open class InputEvent(type: String, eventInitDict: InputEventInit = definedExternally) : UIEvent {\n    open val data: String\n    open val isComposing: Boolean\n\n    companion object {\n        val NONE: Short\n        val CAPTURING_PHASE: Short\n        val AT_TARGET: Short\n        val BUBBLING_PHASE: Short\n    }\n}\n\npublic external interface InputEventInit : UIEventInit {\n    var data: String? /* = \"\" */\n        get() = definedExternally\n        set(value) = definedExternally\n    var isComposing: Boolean? /* = false */\n
```

get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun InputEventInit(data: String? = \"\", isComposing: Boolean? = false, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): InputEventInit {\n val o = js(\"({})\")\n o[\"data\"] = data\n o[\"isComposing\"] = isComposing\n o[\"view\"] = view\n o[\"detail\"] = detail\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [KeyboardEvent](https://developer.mozilla.org/en/docs/Web/API/KeyboardEvent) to Kotlin\n */\npublic external open class KeyboardEvent(type: String, eventInitDict: KeyboardEventInit = definedExternally) : UIEvent {\n open val key: String\n open val code: String\n open val location: Int\n open val ctrlKey: Boolean\n open val shiftKey: Boolean\n open val altKey: Boolean\n open val metaKey: Boolean\n open val repeat: Boolean\n open val isComposing: Boolean\n open val charCode: Int\n open val keyCode: Int\n open val which: Int\n fun getModifierState(keyArg: String): Boolean\n\n companion object {\n val DOM_KEY_LOCATION_STANDARD: Int\n val DOM_KEY_LOCATION_LEFT: Int\n val DOM_KEY_LOCATION_RIGHT: Int\n val DOM_KEY_LOCATION_NUMPAD: Int\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface KeyboardEventInit : EventModifierInit {\n var key: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n var code: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n var location: Int? /* = 0 */\n get() = definedExternally\n set(value) = definedExternally\n var repeat: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n var isComposing: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun KeyboardEventInit(key: String? = \"\", code: String? = \"\", location: Int? = 0, repeat: Boolean? = false, isComposing: Boolean? = false, ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false, metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn: Boolean? = false, modifierFnLock: Boolean? = false, modifierHyper: Boolean? = false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false, modifierSymbol: Boolean? = false, modifierSymbolLock: Boolean? = false, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): KeyboardEventInit {\n val o = js(\"({})\")\n o[\"key\"] = key\n o[\"code\"] = code\n o[\"location\"] = location\n o[\"repeat\"] = repeat\n o[\"isComposing\"] = isComposing\n o[\"ctrlKey\"] = ctrlKey\n o[\"shiftKey\"] = shiftKey\n o[\"altKey\"] = altKey\n o[\"metaKey\"] = metaKey\n o[\"modifierAltGraph\"] = modifierAltGraph\n o[\"modifierCapsLock\"] = modifierCapsLock\n o[\"modifierFn\"] = modifierFn\n o[\"modifierFnLock\"] = modifierFnLock\n o[\"modifierHyper\"] = modifierHyper\n o[\"modifierNumLock\"] = modifierNumLock\n o[\"modifierScrollLock\"] = modifierScrollLock\n o[\"modifierSuper\"] = modifierSuper\n o[\"modifierSymbol\"] = modifierSymbol\n o[\"modifierSymbolLock\"] = modifierSymbolLock\n o[\"view\"] = view\n o[\"detail\"] = detail\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [CompositionEvent](https://developer.mozilla.org/en/docs/Web/API/CompositionEvent) to Kotlin\n */\npublic external open class CompositionEvent(type: String, eventInitDict: CompositionEventInit = definedExternally) : UIEvent {\n open val data: String\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface CompositionEventInit : UIEventInit {\n var data: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun CompositionEventInit(data: String? = \"\", view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): CompositionEventInit {\n val o = js(\"({})\")\n o[\"data\"] = data\n o[\"view\"] = view\n o[\"detail\"] = detail\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript

[Event](https://developer.mozilla.org/en/docs/Web/API/Event) to Kotlin\n */\npublic external open class Event(type: String, eventInitDict: EventInit = definedExternally) {\n   open val type: String\n   open val target: EventTarget?\n   open val currentTarget: EventTarget?\n   open val eventPhase: Short\n   open val bubbles: Boolean\n   open val cancelable: Boolean\n   open val defaultPrevented: Boolean\n   open val composed: Boolean\n   open val isTrusted: Boolean\n   open val timeStamp: Number\n   fun composedPath(): Array<EventTarget>\n   fun stopPropagation()\n   fun stopImmediatePropagation()\n   fun preventDefault()\n   fun initEvent(type: String, bubbles: Boolean, cancelable: Boolean)\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [EventTarget](https://developer.mozilla.org/en/docs/Web/API/EventTarget) to Kotlin\n */\npublic external abstract class EventTarget {\n   fun addEventListener(type: String, callback: EventListener?, options: dynamic = definedExternally)\n   fun addEventListener(type: String, callback: ((Event) -> Unit)?, options: dynamic = definedExternally)\n   fun removeEventListener(type: String, callback: EventListener?, options: dynamic = definedExternally)\n   fun removeEventListener(type: String, callback: ((Event) -> Unit)?, options: dynamic = definedExternally)\n   fun dispatchEvent(event: Event): Boolean\n}\n\n/**\n * Exposes the JavaScript [EventListener](https://developer.mozilla.org/en/docs/Web/API/EventListener) to Kotlin\n */\npublic external interface EventListener {\n   fun handleEvent(event: Event)\n}","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.dom\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.clipboard.*\nimport org.w3c.dom.css.*\nimport org.w3c.dom.encryptedmedia.*\nimport org.w3c.dom.events.*\nimport org.w3c.dom.mediacapture.*\nimport org.w3c.dom.mediasource.*\nimport org.w3c.dom.pointerevents.*\nimport org.w3c.dom.svg.*\nimport org.w3c.fetch.*\nimport org.w3c.files.*\nimport org.w3c.performance.*\nimport org.w3c.workers.*\nimport org.w3c.xhr.*\n\npublic external abstract class HTMLAllCollection {\n   open val length: Int\n   fun item(nameOrIndex: String = definedExternally): UnionElementOrHTMLCollection?\n   fun namedItem(name: String): UnionElementOrHTMLCollection?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun HTMLAllCollection.get(index: Int): Element? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun HTMLAllCollection.get(name: String): UnionElementOrHTMLCollection? = asDynamic()[name]\n\n/**\n * Exposes the JavaScript [HTMLFormControlsCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLFormControlsCollection) to Kotlin\n */\npublic external abstract class HTMLFormControlsCollection : HTMLCollection\n\n/**\n * Exposes the JavaScript [RadioNodeList](https://developer.mozilla.org/en/docs/Web/API/RadioNodeList) to Kotlin\n */\npublic external abstract class RadioNodeList : NodeList, UnionElementOrRadioNodeList {\n   open var value: String\n}\n\n/**\n * Exposes the JavaScript [HTMLOptionsCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLOptionsCollection) to Kotlin\n */\npublic external abstract class HTMLOptionsCollection : HTMLCollection {\n   override var length: Int\n   open var selectedIndex: Int\n   fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)\n   fun remove(index: Int)\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun HTMLOptionsCollection.set(index: Int, option: HTMLOptionElement?) { asDynamic()[index] = option }\n\n/**\n * Exposes the JavaScript [HTMLElement](https://developer.mozilla.org/en/docs/Web/API/HTMLElement) to Kotlin\n */\npublic external abstract class HTMLElement : Element, GlobalEventHandlers, DocumentAndElementEventHandlers, ElementContentEditable, ElementCSSInlineStyle {\n   open var title: String\n   open var lang: String\n   open var translate: Boolean\n   open var dir: String\n   open val dataset:

DOMStringMap\n   open var hidden: Boolean\n   open var tabIndex: Int\n   open var accessKey: String\n   open val accessKeyLabel: String\n   open var draggable: Boolean\n   open val dropzone: DOMTokenList\n   open var contextMenu: HTMLMenuElement?\n   open var spellcheck: Boolean\n   open var innerText: String\n   open val offsetParent: Element?\n   open val offsetTop: Int\n   open val offsetLeft: Int\n   open val offsetWidth: Int\n   open val offsetHeight: Int\n   fun click()\n   fun focus()\n   fun blur()\n   fun forceSpellCheck()\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [HTMLUnknownElement](https://developer.mozilla.org/en/docs/Web/API/HTMLUnknownElement) to Kotlin\n */\npublic external abstract class HTMLUnknownElement : HTMLElement {\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [DOMStringMap](https://developer.mozilla.org/en/docs/Web/API/DOMStringMap) to Kotlin\n */\npublic external abstract class DOMStringMap\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun DOMStringMap.get(name: String): String? = asDynamic()[name]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun DOMStringMap.set(name: String, value: String) { asDynamic()[name] = value }\n\n/**\n * Exposes the JavaScript [HTMLHtmlElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHtmlElement) to Kotlin\n */\npublic external abstract class HTMLHtmlElement : HTMLElement {\n   open var version: String\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [HTMLHeadElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadElement) to Kotlin\n */\npublic external abstract class HTMLHeadElement : HTMLElement {\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val

DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLTitleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTitleElement) to Kotlin\n */\npublic external abstract class HTMLTitleElement : HTMLElement {\n open var text: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLBaseElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBaseElement) to Kotlin\n */\npublic external abstract class HTMLBaseElement : HTMLElement {\n open var href: String\n open var target: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLLinkElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLinkElement) to Kotlin\n */\npublic external abstract class HTMLLinkElement : HTMLElement, LinkStyle {\n open var href: String\n open var crossOrigin: String?\n open var rel: String\n open var `as`: RequestDestination\n open val relList: DOMTokenList\n open var media: String\n open var nonce: String\n open var hreflang: String\n open var type: String\n open val sizes: DOMTokenList\n open var referrerPolicy: String\n open var charset: String\n open var rev: String\n open var target: String\n open var scope: String\n open var workerType: WorkerType\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLMetaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMetaElement) to Kotlin\n */\npublic external abstract class HTMLMetaElement : HTMLElement {\n open var name: String\n open var httpEquiv: String\n open var content: String\n open var scheme: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE:

Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLStyleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLStyleElement) to Kotlin\n */\npublic external abstract class HTMLStyleElement : HTMLElement, LinkStyle {\n open var media: String\n open var nonce: String\n open var type: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLBodyElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBodyElement) to Kotlin\n */\npublic external abstract class HTMLBodyElement : HTMLElement, WindowEventHandlers {\n open var text: String\n open var link: String\n open var vLink: String\n open var aLink: String\n open var bgColor: String\n open var background: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLHeadingElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadingElement) to Kotlin\n */\npublic external abstract class HTMLHeadingElement : HTMLElement {\n open var align: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLParagraphElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParagraphElement) to Kotlin\n */\npublic external abstract class HTMLParagraphElement : HTMLElement {\n open var align: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val

COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLHRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHRElement) to Kotlin\n */\npublic external abstract class HTMLHRElement : HTMLElement {\n open var align: String\n open var color: String\n open var noShade: Boolean\n open var size: String\n open var width: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLPreElement](https://developer.mozilla.org/en/docs/Web/API/HTMLPreElement) to Kotlin\n */\npublic external abstract class HTMLPreElement : HTMLElement {\n open var width: Int\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLQuoteElement](https://developer.mozilla.org/en/docs/Web/API/HTMLQuoteElement) to Kotlin\n */\npublic external abstract class HTMLQuoteElement : HTMLElement {\n open var cite: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLOListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOListElement) to Kotlin\n */\npublic external abstract class HTMLOListElement : HTMLElement {\n open var reversed: Boolean\n open var start: Int\n open var type: String\n open var compact: Boolean\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val

DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLUListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLUListElement) to Kotlin\n */\npublic external abstract class HTMLUListElement : HTMLElement {\n open var compact: Boolean\n open var type: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLLIElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLIElement) to Kotlin\n */\npublic external abstract class HTMLLIElement : HTMLElement {\n open var value: Int\n open var type: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLDListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDListElement) to Kotlin\n */\npublic external abstract class HTMLDListElement : HTMLElement {\n open var compact: Boolean\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLDivElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDivElement) to Kotlin\n */\npublic external abstract class HTMLDivElement : HTMLElement {\n open var align: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript

[HTMLAnchorElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAnchorElement) to Kotlin\n */\npublic external abstract class HTMLAnchorElement : HTMLElement, HTMLHyperlinkElementUtils {\n open var target: String\n open var download: String\n open var ping: String\n open var rel: String\n open val relList: DOMTokenList\n open var hreflang: String\n open var type: String\n open var text: String\n open var referrerPolicy: String\n open var coords: String\n open var charset: String\n open var name: String\n open var rev: String\n open var shape: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLDataElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataElement) to Kotlin\n */\npublic external abstract class HTMLDataElement : HTMLElement {\n open var value: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLTimeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTimeElement) to Kotlin\n */\npublic external abstract class HTMLTimeElement : HTMLElement {\n open var dateTime: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLSpanElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSpanElement) to Kotlin\n */\npublic external abstract class HTMLSpanElement : HTMLElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript

[HTMLBRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBRElement) to Kotlin\n */\npublic external abstract class HTMLBRElement : HTMLElement {\n open var clear: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLHyperlinkElementUtils](https://developer.mozilla.org/en/docs/Web/API/HTMLHyperlinkElementUtils) to Kotlin\n */\npublic external interface HTMLHyperlinkElementUtils {\n var href: String\n val origin: String\n var protocol: String\n var username: String\n var password: String\n var host: String\n var hostname: String\n var port: String\n var pathname: String\n var search: String\n var hash: String\n}\n\n/**\n * Exposes the JavaScript [HTMLModElement](https://developer.mozilla.org/en/docs/Web/API/HTMLModElement) to Kotlin\n */\npublic external abstract class HTMLModElement : HTMLElement {\n open var cite: String\n open var dateTime: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLPictureElement](https://developer.mozilla.org/en/docs/Web/API/HTMLPictureElement) to Kotlin\n */\npublic external abstract class HTMLPictureElement : HTMLElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLSourceElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSourceElement) to Kotlin\n */\npublic external abstract class HTMLSourceElement : HTMLElement {\n open var src: String\n open var type: String\n open var srcset: String\n open var sizes: String\n open var media: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n

val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLImageElement](https://developer.mozilla.org/en/docs/Web/API/HTMLImageElement) to Kotlin\n */\npublic external abstract class HTMLImageElement : HTMLElement, HTMLOrSVGImageElement, TexImageSource {\n open var alt: String\n open var src: String\n open var srcset: String\n open var sizes: String\n open var crossOrigin: String?\n open var useMap: String\n open var isMap: Boolean\n open var width: Int\n open var height: Int\n open val naturalWidth: Int\n open val naturalHeight: Int\n open val complete: Boolean\n open val currentSrc: String\n open var referrerPolicy: String\n open var name: String\n open var lowsrc: String\n open var align: String\n open var hspace: Int\n open var vspace: Int\n open var longDesc: String\n open var border: String\n open val x: Int\n open val y: Int\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLIFrameElement](https://developer.mozilla.org/en/docs/Web/API/HTMLIFrameElement) to Kotlin\n */\npublic external abstract class HTMLIFrameElement : HTMLElement {\n open var src: String\n open var srcdoc: String\n open var name: String\n open val sandbox: DOMTokenList\n open var allowFullscreen: Boolean\n open var allowUserMedia: Boolean\n open var width: String\n open var height: String\n open var referrerPolicy: String\n open val contentDocument: Document?\n open val contentWindow: Window?\n open var align: String\n open var scrolling: String\n open var frameBorder: String\n open var longDesc: String\n open var marginHeight: String\n open var marginWidth: String\n fun getSVGDocument(): Document?\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLEmbedElement](https://developer.mozilla.org/en/docs/Web/API/HTMLEmbedElement) to Kotlin\n */\npublic external abstract class HTMLEmbedElement : HTMLElement {\n open var src: String\n open var type: String\n open var width: String\n open var height: String\n open var align: String\n open var name: String\n fun getSVGDocument(): Document?\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript

[HTMLObjectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLObjectElement) to Kotlin\n */\npublic external abstract class HTMLObjectElement : HTMLElement {\n open var data: String\n open var type: String\n open var typeMustMatch: Boolean\n open var name: String\n open var useMap: String\n open val form: HTMLFormElement?\n open var width: String\n open var height: String\n open val contentDocument: Document?\n open val contentWindow: Window?\n open val willValidate: Boolean\n open val validity: ValidityState\n open val validationMessage: String\n open var align: String\n open var archive: String\n open var code: String\n open var declare: Boolean\n open var hspace: Int\n open var standby: String\n open var vspace: Int\n open var codeBase: String\n open var codeType: String\n open var border: String\n fun getSVGDocument(): Document?\n fun checkValidity(): Boolean\n fun reportValidity(): Boolean\n fun setCustomValidity(error: String)\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLParamElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParamElement) to Kotlin\n */\npublic external abstract class HTMLParamElement : HTMLElement {\n open var name: String\n open var value: String\n open var type: String\n open var valueType: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLVideoElement](https://developer.mozilla.org/en/docs/Web/API/HTMLVideoElement) to Kotlin\n */\npublic external abstract class HTMLVideoElement : HTMLMediaElement, CanvasImageSource, TexImageSource {\n open var width: Int\n open var height: Int\n open val videoWidth: Int\n open val videoHeight: Int\n open var poster: String\n open var playsInline: Boolean\n\n companion object {\n val NETWORK_EMPTY: Short\n val NETWORK_IDLE: Short\n val NETWORK_LOADING: Short\n val NETWORK_NO_SOURCE: Short\n val HAVE_NOTHING: Short\n val HAVE_METADATA: Short\n val HAVE_CURRENT_DATA: Short\n val HAVE_FUTURE_DATA: Short\n val HAVE_ENOUGH_DATA: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLAudioElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAudioElement) to Kotlin\n

*/\npublic external abstract class HTMLAudioElement : HTMLMediaElement {\n companion object {\n val NETWORK_EMPTY: Short\n val NETWORK_IDLE: Short\n val NETWORK_LOADING: Short\n val NETWORK_NO_SOURCE: Short\n val HAVE_NOTHING: Short\n val HAVE_METADATA: Short\n val HAVE_CURRENT_DATA: Short\n val HAVE_FUTURE_DATA: Short\n val HAVE_ENOUGH_DATA: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLTrackElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTrackElement) to Kotlin\n */\npublic external abstract class HTMLTrackElement : HTMLElement {\n open var kind: String\n open var src: String\n open var srclang: String\n open var label: String\n open var default: Boolean\n open val readyState: Short\n open val track: TextTrack\n\n companion object {\n val NONE: Short\n val LOADING: Short\n val LOADED: Short\n val ERROR: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLMediaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMediaElement) to Kotlin\n */\npublic external abstract class HTMLMediaElement : HTMLElement {\n open val error: MediaError?\n open var src: String\n open var srcObject: MediaProvider?\n open val currentSrc: String\n open var crossOrigin: String?\n open val networkState: Short\n open var preload: String\n open val buffered: TimeRanges\n open val readyState: Short\n open val seeking: Boolean\n open var currentTime: Double\n open val duration: Double\n open val paused: Boolean\n open var defaultPlaybackRate: Double\n open var playbackRate: Double\n open val played: TimeRanges\n open val seekable: TimeRanges\n open val ended: Boolean\n open var autoplay: Boolean\n open var loop: Boolean\n open var controls: Boolean\n open var volume: Double\n open var muted: Boolean\n open var defaultMuted: Boolean\n open val audioTracks: AudioTrackList\n open val videoTracks: VideoTrackList\n open val textTracks: TextTrackList\n open val mediaKeys: MediaKeys?\n open var onencrypted: ((Event) -> dynamic)?\n open var onwaitingforkey: ((Event) -> dynamic)?\n fun load()\n fun canPlayType(type: String): CanPlayTypeResult\n fun fastSeek(time: Double)\n fun getStartDate(): dynamic\n fun play(): Promise<Unit>\n fun pause()\n fun addTextTrack(kind: TextTrackKind, label: String = definedExternally, language: String = definedExternally): TextTrack\n fun setMediaKeys(mediaKeys: MediaKeys?): Promise<Unit>\n\n companion object {\n val NETWORK_EMPTY: Short\n val NETWORK_IDLE: Short\n val NETWORK_LOADING: Short\n val NETWORK_NO_SOURCE: Short\n val HAVE_NOTHING: Short\n val HAVE_METADATA: Short\n val HAVE_CURRENT_DATA: Short\n val HAVE_FUTURE_DATA: Short\n val HAVE_ENOUGH_DATA: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val

DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [MediaError](https://developer.mozilla.org/en/docs/Web/API/MediaError) to Kotlin\n */\npublic external abstract class MediaError {\n open val code: Short\n\n companion object {\n val MEDIA_ERR_ABORTED: Short\n val MEDIA_ERR_NETWORK: Short\n val MEDIA_ERR_DECODE: Short\n val MEDIA_ERR_SRC_NOT_SUPPORTED: Short\n }\n}\n\n/**\n * Exposes the JavaScript [AudioTrackList](https://developer.mozilla.org/en/docs/Web/API/AudioTrackList) to Kotlin\n */\npublic external abstract class AudioTrackList : EventTarget {\n open val length: Int\n open var onchange: ((Event) -> dynamic)?\n open var onaddtrack: ((TrackEvent) -> dynamic)?\n open var onremovetrack: ((TrackEvent) -> dynamic)?\n fun getTrackById(id: String): AudioTrack?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun AudioTrackList.get(index: Int): AudioTrack? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript [AudioTrack](https://developer.mozilla.org/en/docs/Web/API/AudioTrack) to Kotlin\n */\npublic external abstract class AudioTrack : UnionAudioTrackOrTextTrackOrVideoTrack {\n open val id: String\n open val kind: String\n open val label: String\n open val language: String\n open var enabled: Boolean\n open val sourceBuffer: SourceBuffer?\n}\n\n/**\n * Exposes the JavaScript [VideoTrackList](https://developer.mozilla.org/en/docs/Web/API/VideoTrackList) to Kotlin\n */\npublic external abstract class VideoTrackList : EventTarget {\n open val length: Int\n open val selectedIndex: Int\n open var onchange: ((Event) -> dynamic)?\n open var onaddtrack: ((TrackEvent) -> dynamic)?\n open var onremovetrack: ((TrackEvent) -> dynamic)?\n fun getTrackById(id: String): VideoTrack?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun VideoTrackList.get(index: Int): VideoTrack? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript [VideoTrack](https://developer.mozilla.org/en/docs/Web/API/VideoTrack) to Kotlin\n */\npublic external abstract class VideoTrack : UnionAudioTrackOrTextTrackOrVideoTrack {\n open val id: String\n open val kind: String\n open val label: String\n open val language: String\n open var selected: Boolean\n open val sourceBuffer: SourceBuffer?\n}\n\npublic external abstract class TextTrackList : EventTarget {\n open val length: Int\n open var onchange: ((Event) -> dynamic)?\n open var onaddtrack: ((TrackEvent) -> dynamic)?\n open var onremovetrack: ((TrackEvent) -> dynamic)?\n fun getTrackById(id: String): TextTrack?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun TextTrackList.get(index: Int): TextTrack? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript [TextTrack](https://developer.mozilla.org/en/docs/Web/API/TextTrack) to Kotlin\n */\npublic external abstract class TextTrack : EventTarget, UnionAudioTrackOrTextTrackOrVideoTrack {\n open val kind: TextTrackKind\n open val label: String\n open val language: String\n open val id: String\n open val inBandMetadataTrackDispatchType: String\n open var mode: TextTrackMode\n open val cues: TextTrackCueList?\n open val activeCues: TextTrackCueList?\n open var oncuechange: ((Event) -> dynamic)?\n open val sourceBuffer: SourceBuffer?\n fun addCue(cue: TextTrackCue)\n fun removeCue(cue: TextTrackCue)\n}\n\npublic external abstract class TextTrackCueList {\n open val length: Int\n fun getCueById(id: String): TextTrackCue?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun TextTrackCueList.get(index: Int): TextTrackCue? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript [TextTrackCue](https://developer.mozilla.org/en/docs/Web/API/TextTrackCue) to Kotlin\n */\npublic external abstract class TextTrackCue : EventTarget {\n open val track: TextTrack?\n open var id: String\n open var

```
startTime: Double\n    open var endTime: Double\n    open var pauseOnExit: Boolean\n    open var onenter: ((Event) -> dynamic)?\n    open var onexit: ((Event) -> dynamic)?\n}\n\n/**\n * Exposes the JavaScript [TimeRanges](https://developer.mozilla.org/en/docs/Web/API/TimeRanges) to Kotlin\n */\npublic external abstract class TimeRanges {\n    open val length: Int\n    fun start(index: Int): Double\n    fun end(index: Int): Double\n}\n\n/**\n * Exposes the JavaScript [TrackEvent](https://developer.mozilla.org/en/docs/Web/API/TrackEvent) to Kotlin\n */\npublic external open class TrackEvent(type: String, eventInitDict: TrackEventInit = definedExternally) : Event {\n    open val track: UnionAudioTrackOrTextTrackOrVideoTrack?\n\n    companion object {\n        val NONE: Short\n        val CAPTURING_PHASE: Short\n        val AT_TARGET: Short\n        val BUBBLING_PHASE: Short\n    }\n}\n\npublic external interface TrackEventInit : EventInit {\n    var track: UnionAudioTrackOrTextTrackOrVideoTrack? /* = null */\n        get() = definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun TrackEventInit(track: UnionAudioTrackOrTextTrackOrVideoTrack? = null, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): TrackEventInit {\n    val o = js(\"({})\")\n    o[\"track\"] = track\n    o[\"bubbles\"] = bubbles\n    o[\"cancelable\"] = cancelable\n    o[\"composed\"] = composed\n    return o\n}\n\n/**\n * Exposes the JavaScript [HTMLMapElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMapElement) to Kotlin\n */\npublic external abstract class HTMLMapElement : HTMLElement {\n    open var name: String\n    open val areas: HTMLCollection\n\n    companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n        val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n        val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript [HTMLAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAreaElement) to Kotlin\n */\npublic external abstract class HTMLAreaElement : HTMLElement, HTMLHyperlinkElementUtils {\n    open var alt: String\n    open var coords: String\n    open var shape: String\n    open var target: String\n    open var download: String\n    open var ping: String\n    open var rel: String\n    open val relList: DOMTokenList\n    open var referrerPolicy: String\n    open var noHref: Boolean\n\n    companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n        val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n        val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript [HTMLTableElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableElement) to Kotlin\n */\npublic external abstract class HTMLTableElement : HTMLElement {\n    open var caption: HTMLTableCaptionElement?\n    open var tHead: HTMLTableSectionElement?\n    open var tFoot: HTMLTableSectionElement?\n    open val tBodies: HTMLCollection\n    open val rows: HTMLCollection\n    open var align: String\n    open var border: String\n    open var frame: String\n    open var rules: String\n    open var summary: String\n    open var width: String\n    open var bgColor: String\n    open var cellPadding: String\n    open
```

var cellSpacing: String\n   fun createCaption(): HTMLTableCaptionElement\n   fun deleteCaption()\n   fun createTHead(): HTMLTableSectionElement\n   fun deleteTHead()\n   fun createTFoot(): HTMLTableSectionElement\n   fun deleteTFoot()\n   fun createTBody(): HTMLTableSectionElement\n   fun insertRow(index: Int = definedExternally): HTMLTableRowElement\n   fun deleteRow(index: Int)\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [HTMLTableCaptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCaptionElement) to Kotlin\n */\npublic external abstract class HTMLTableCaptionElement : HTMLElement {\n   open var align: String\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [HTMLTableColElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableColElement) to Kotlin\n */\npublic external abstract class HTMLTableColElement : HTMLElement {\n   open var span: Int\n   open var align: String\n   open var ch: String\n   open var chOff: String\n   open var vAlign: String\n   open var width: String\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [HTMLTableSectionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableSectionElement) to Kotlin\n */\npublic external abstract class HTMLTableSectionElement : HTMLElement {\n   open val rows: HTMLCollection\n   open var align: String\n   open var ch: String\n   open var chOff: String\n   open var vAlign: String\n   fun insertRow(index: Int = definedExternally): HTMLElement\n   fun deleteRow(index: Int)\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLTableRowElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableRowElement) to Kotlin\n */\npublic external abstract class HTMLTableRowElement : HTMLElement {\n open val rowIndex: Int\n open val sectionRowIndex: Int\n open val cells: HTMLCollection\n open var align: String\n open var ch: String\n open var chOff: String\n open var vAlign: String\n open var bgColor: String\n fun insertCell(index: Int = definedExternally): HTMLElement\n fun deleteCell(index: Int)\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLTableCellElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCellElement) to Kotlin\n */\npublic external abstract class HTMLTableCellElement : HTMLElement {\n open var colSpan: Int\n open var rowSpan: Int\n open var headers: String\n open val cellIndex: Int\n open var scope: String\n open var abbr: String\n open var align: String\n open var axis: String\n open var height: String\n open var width: String\n open var ch: String\n open var chOff: String\n open var noWrap: Boolean\n open var vAlign: String\n open var bgColor: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLFormElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFormElement) to Kotlin\n */\npublic external abstract class HTMLFormElement : HTMLElement {\n open var acceptCharset: String\n open var action: String\n open var autocomplete: String\n open var enctype: String\n open var encoding: String\n open var method: String\n open var name: String\n open var noValidate: Boolean\n open var target: String\n open val elements: HTMLFormControlsCollection\n open val length: Int\n fun submit()\n fun reset()\n fun checkValidity(): Boolean\n fun reportValidity(): Boolean\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun HTMLFormElement.get(index: Int): Element? =

asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun HTMLFormElement.get(name: String): UnionElementOrRadioNodeList? = asDynamic()[name]\n\n/**\n * Exposes the JavaScript [HTMLLabelElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLabelElement) to Kotlin\n */\npublic external abstract class HTMLLabelElement : HTMLElement {\n open val form: HTMLFormElement?\n open var htmlFor: String\n open val control: HTMLElement?\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLInputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLInputElement) to Kotlin\n */\npublic external abstract class HTMLInputElement : HTMLElement {\n open var accept: String\n open var alt: String\n open var autocomplete: String\n open var autofocus: Boolean\n open var defaultChecked: Boolean\n open var checked: Boolean\n open var dirName: String\n open var disabled: Boolean\n open val form: HTMLFormElement?\n open val files: FileList?\n open var formAction: String\n open var formEnctype: String\n open var formMethod: String\n open var formNoValidate: Boolean\n open var formTarget: String\n open var height: Int\n open var indeterminate: Boolean\n open var inputMode: String\n open val list: HTMLElement?\n open var max: String\n open var maxLength: Int\n open var min: String\n open var minLength: Int\n open var multiple: Boolean\n open var name: String\n open var pattern: String\n open var placeholder: String\n open var readOnly: Boolean\n open var required: Boolean\n open var size: Int\n open var src: String\n open var step: String\n open var type: String\n open var defaultValue: String\n open var value: String\n open var valueAsDate: dynamic\n open var valueAsNumber: Double\n open var width: Int\n open val willValidate: Boolean\n open val validity: ValidityState\n open val validationMessage: String\n open val labels: NodeList\n open var selectionStart: Int?\n open var selectionEnd: Int?\n open var selectionDirection: String?\n open var align: String\n open var useMap: String\n fun stepUp(n: Int = definedExternally)\n fun stepDown(n: Int = definedExternally)\n fun checkValidity(): Boolean\n fun reportValidity(): Boolean\n fun setCustomValidity(error: String)\n fun select()\n fun setRangeText(replacement: String)\n fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally)\n fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally)\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLButtonElement](https://developer.mozilla.org/en/docs/Web/API/HTMLButtonElement) to Kotlin\n */\npublic external abstract class HTMLButtonElement : HTMLElement {\n open var autofocus: Boolean\n open var disabled: Boolean\n open val form: HTMLFormElement?\n open var formAction: String\n open var formEnctype: String\n open var formMethod: String\n open var formNoValidate: Boolean\n open var

formTarget: String\n   open var name: String\n   open var type: String\n   open var value: String\n   open var menu: HTMLMenuElement?\n   open val willValidate: Boolean\n   open val validity: ValidityState\n   open val validationMessage: String\n   open val labels: NodeList\n   fun checkValidity(): Boolean\n   fun reportValidity(): Boolean\n   fun setCustomValidity(error: String)\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [HTMLSelectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSelectElement) to Kotlin\n */\npublic external abstract class HTMLSelectElement : HTMLElement, ItemArrayLike<Element> {\n   open var autocomplete: String\n   open var autofocus: Boolean\n   open var disabled: Boolean\n   open val form: HTMLFormElement?\n   open var multiple: Boolean\n   open var name: String\n   open var required: Boolean\n   open var size: Int\n   open val type: String\n   open val options: HTMLOptionsCollection\n   override var length: Int\n   open val selectedOptions: HTMLCollection\n   open var selectedIndex: Int\n   open var value: String\n   open val willValidate: Boolean\n   open val validity: ValidityState\n   open val validationMessage: String\n   open val labels: NodeList\n   fun namedItem(name: String): HTMLOptionElement?\n   fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)\n   fun remove(index: Int)\n   fun checkValidity(): Boolean\n   fun reportValidity(): Boolean\n   fun setCustomValidity(error: String)\n   override fun item(index: Int): Element?\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun HTMLSelectElement.get(index: Int): Element? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun HTMLSelectElement.set(index: Int, option: HTMLOptionElement?) { asDynamic()[index] = option }\n\n/**\n * Exposes the JavaScript [HTMLDataListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataListElement) to Kotlin\n */\npublic external abstract class HTMLDataListElement : HTMLElement {\n   open val options: HTMLCollection\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val

DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLOptGroupElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptGroupElement) to Kotlin\n */\npublic external abstract class HTMLOptGroupElement : HTMLElement, UnionHTMLOptGroupElementOrHTMLOptionElement {\n open var disabled: Boolean\n open var label: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLOptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptionElement) to Kotlin\n */\npublic external abstract class HTMLOptionElement : HTMLElement, UnionHTMLOptGroupElementOrHTMLOptionElement {\n open var disabled: Boolean\n open val form: HTMLFormElement?\n open var label: String\n open var defaultSelected: Boolean\n open var selected: Boolean\n open var value: String\n open var text: String\n open val index: Int\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLTextAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTextAreaElement) to Kotlin\n */\npublic external abstract class HTMLTextAreaElement : HTMLElement {\n open var autocomplete: String\n open var autofocus: Boolean\n open var cols: Int\n open var dirName: String\n open var disabled: Boolean\n open val form: HTMLFormElement?\n open var inputMode: String\n open var maxLength: Int\n open var minLength: Int\n open var name: String\n open var placeholder: String\n open var readOnly: Boolean\n open var required: Boolean\n open var rows: Int\n open var wrap: String\n open val type: String\n open var defaultValue: String\n open var value: String\n open val textLength: Int\n open val willValidate: Boolean\n open val validity: ValidityState\n open val validationMessage: String\n open val labels: NodeList\n open var selectionStart: Int?\n open var selectionEnd: Int?\n open var selectionDirection: String?\n fun checkValidity(): Boolean\n fun reportValidity(): Boolean\n fun setCustomValidity(error: String)\n fun select()\n fun setRangeText(replacement: String)\n fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally)\n fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally)\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val

DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLKeygenElement](https://developer.mozilla.org/en/docs/Web/API/HTMLKeygenElement) to Kotlin\n */\npublic external abstract class HTMLKeygenElement : HTMLElement {\n open var autofocus: Boolean\n open var challenge: String\n open var disabled: Boolean\n open val form: HTMLFormElement?\n open var keytype: String\n open var name: String\n open val type: String\n open val willValidate: Boolean\n open val validity: ValidityState\n open val validationMessage: String\n open val labels: NodeList\n fun checkValidity(): Boolean\n fun reportValidity(): Boolean\n fun setCustomValidity(error: String)\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLOutputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOutputElement) to Kotlin\n */\npublic external abstract class HTMLOutputElement : HTMLElement {\n open val htmlFor: DOMTokenList\n open val form: HTMLFormElement?\n open var name: String\n open val type: String\n open var defaultValue: String\n open var value: String\n open val willValidate: Boolean\n open val validity: ValidityState\n open val validationMessage: String\n open val labels: NodeList\n fun checkValidity(): Boolean\n fun reportValidity(): Boolean\n fun setCustomValidity(error: String)\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLProgressElement](https://developer.mozilla.org/en/docs/Web/API/HTMLProgressElement) to Kotlin\n */\npublic external abstract class HTMLProgressElement : HTMLElement {\n open var value: Double\n open var max: Double\n open val position: Double\n open val labels: NodeList\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLMeterElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMeterElement) to Kotlin\n */\npublic external abstract class HTMLMeterElement : HTMLElement {\n open var value: Double\n open var min:

Double\n   open var max: Double\n   open var low: Double\n   open var high: Double\n   open var optimum: Double\n   open val labels: NodeList\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [HTMLFieldSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFieldSetElement) to Kotlin\n */\npublic external abstract class HTMLFieldSetElement : HTMLElement {\n   open var disabled: Boolean\n   open val form: HTMLFormElement?\n   open var name: String\n   open val type: String\n   open val elements: HTMLCollection\n   open val willValidate: Boolean\n   open val validity: ValidityState\n   open val validationMessage: String\n   fun checkValidity(): Boolean\n   fun reportValidity(): Boolean\n   fun setCustomValidity(error: String)\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [HTMLLegendElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLegendElement) to Kotlin\n */\npublic external abstract class HTMLLegendElement : HTMLElement {\n   open val form: HTMLFormElement?\n   open var align: String\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [ValidityState](https://developer.mozilla.org/en/docs/Web/API/ValidityState) to Kotlin\n */\npublic external abstract class ValidityState {\n   open val valueMissing: Boolean\n   open val typeMismatch: Boolean\n   open val patternMismatch: Boolean\n   open val tooLong: Boolean\n   open val tooShort: Boolean\n   open val rangeUnderflow: Boolean\n   open val rangeOverflow: Boolean\n   open val stepMismatch: Boolean\n   open val badInput: Boolean\n   open val customError: Boolean\n   open val valid: Boolean\n}\n\n/**\n * Exposes the JavaScript [HTMLDetailsElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDetailsElement) to Kotlin\n */\npublic external abstract class HTMLDetailsElement : HTMLElement {\n   open var open: Boolean\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val

COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external abstract class HTMLMenuElement : HTMLElement {\n open var type: String\n open var label: String\n open var compact: Boolean\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external abstract class HTMLMenuItemElement : HTMLElement {\n open var type: String\n open var label: String\n open var icon: String\n open var disabled: Boolean\n open var checked: Boolean\n open var radiogroup: String\n open var default: Boolean\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external open class RelatedEvent(type: String, eventInitDict: RelatedEventInit = definedExternally) : Event {\n open val relatedTarget: EventTarget?\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface RelatedEventInit : EventInit {\n var relatedTarget: EventTarget? /* = null */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun RelatedEventInit(relatedTarget: EventTarget? = null, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): RelatedEventInit {\n val o = js(\"({})\")\n o[\"relatedTarget\"] = relatedTarget\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [HTMLDialogElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDialogElement) to Kotlin\n */\npublic external abstract class HTMLDialogElement : HTMLElement {\n open var open: Boolean\n open var returnValue: String\n fun show(anchor: UnionElementOrMouseEvent = definedExternally)\n fun showModal(anchor: UnionElementOrMouseEvent = definedExternally)\n fun close(returnValue: String = definedExternally)\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n

```
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[HTMLScriptElement](https://developer.mozilla.org/en/docs/Web/API/HTMLScriptElement) to Kotlin\n */\npublic
external abstract class HTMLScriptElement : HTMLElement, HTMLOrSVGScriptElement {\n    open var src:
String\n    open var type: String\n    open var charset: String\n    open var async: Boolean\n    open var defer:
Boolean\n    open var crossOrigin: String?\n    open var text: String\n    open var nonce: String\n    open var event:
String\n    open var htmlFor: String\n\n    companion object {\n        val ELEMENT_NODE: Short\n        val
ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val
ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE: Short\n        val
PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[HTMLTemplateElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTemplateElement) to Kotlin\n
*/\npublic external abstract class HTMLTemplateElement : HTMLElement {\n    open val content:
DocumentFragment\n\n    companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE:
Short\n        val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val
ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE: Short\n        val
PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[HTMLSlotElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSlotElement) to Kotlin\n */\npublic
external abstract class HTMLSlotElement : HTMLElement {\n    open var name: String\n    fun
assignedNodes(options: AssignedNodesOptions = definedExternally): Array<Node>\n\n    companion object {\n
val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val
CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE:
Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\npublic external interface
AssignedNodesOptions {\n    var flatten: Boolean? /* = false */\n        get() = definedExternally\n        set(value) =
definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun AssignedNodesOptions(flatten:
Boolean? = false): AssignedNodesOptions {\n    val o = js(\"({})\")\n    o[\"flatten\"] = flatten\n    return
o\n}\n\n/**\n * Exposes the JavaScript
[HTMLCanvasElement](https://developer.mozilla.org/en/docs/Web/API/HTMLCanvasElement) to Kotlin\n
*/\npublic external abstract class HTMLCanvasElement : HTMLElement, CanvasImageSource, TexImageSource
```

{\n   open var width: Int\n   open var height: Int\n   fun getContext(contextId: String, vararg arguments: Any?): RenderingContext?\n   fun toDataURL(type: String = definedExternally, quality: Any? = definedExternally): String\n   fun toBlob(_callback: (Blob?) -> Unit, type: String = definedExternally, quality: Any? = definedExternally)\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\npublic external interface CanvasRenderingContext2DSettings {\n   var alpha: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun CanvasRenderingContext2DSettings(alpha: Boolean? = true): CanvasRenderingContext2DSettings {\n   val o = js(\"({})\")\n   o[\"alpha\"] = alpha\n   return o\n}\n\n/**\n * Exposes the JavaScript [CanvasRenderingContext2D](https://developer.mozilla.org/en/docs/Web/API/CanvasRenderingContext2D) to Kotlin\n */\npublic external abstract class CanvasRenderingContext2D : CanvasState, CanvasTransform, CanvasCompositing, CanvasImageSmoothing, CanvasFillStrokeStyles, CanvasShadowStyles, CanvasFilters, CanvasRect, CanvasDrawPath, CanvasUserInterface, CanvasText, CanvasDrawImage, CanvasHitRegion, CanvasImageData, CanvasPathDrawingStyles, CanvasTextDrawingStyles, CanvasPath, RenderingContext {\n   open val canvas: HTMLCanvasElement\n}\n\npublic external interface CanvasState {\n   fun save()\n   fun restore()\n}\n\npublic external interface CanvasTransform {\n   fun scale(x: Double, y: Double)\n   fun rotate(angle: Double)\n   fun translate(x: Double, y: Double)\n   fun transform(a: Double, b: Double, c: Double, d: Double, e: Double, f: Double)\n   fun getTransform(): DOMMatrix\n   fun setTransform(a: Double, b: Double, c: Double, d: Double, e: Double, f: Double)\n   fun setTransform(transform: dynamic = definedExternally)\n   fun resetTransform()\n}\n\npublic external interface CanvasCompositing {\n   var globalAlpha: Double\n   var globalCompositeOperation: String\n}\n\npublic external interface CanvasImageSmoothing {\n   var imageSmoothingEnabled: Boolean\n   var imageSmoothingQuality: ImageSmoothingQuality\n}\n\npublic external interface CanvasFillStrokeStyles {\n   var strokeStyle: dynamic\n   get() = definedExternally\n   set(value) = definedExternally\n   var fillStyle: dynamic\n   get() = definedExternally\n   set(value) = definedExternally\n   fun createLinearGradient(x0: Double, y0: Double, x1: Double, y1: Double): CanvasGradient\n   fun createRadialGradient(x0: Double, y0: Double, r0: Double, x1: Double, y1: Double, r1: Double): CanvasGradient\n   fun createPattern(image: CanvasImageSource, repetition: String): CanvasPattern?\n}\n\npublic external interface CanvasShadowStyles {\n   var shadowOffsetX: Double\n   var shadowOffsetY: Double\n   var shadowBlur: Double\n   var shadowColor: String\n}\n\npublic external interface CanvasFilters {\n   var filter: String\n}\n\npublic external interface CanvasRect {\n   fun clearRect(x: Double, y: Double, w: Double, h: Double)\n   fun fillRect(x: Double, y: Double, w: Double, h: Double)\n   fun strokeRect(x: Double, y: Double, w: Double, h: Double)\n}\n\npublic external interface CanvasDrawPath {\n   fun beginPath()\n   fun fill(fillRule: CanvasFillRule = definedExternally)\n   fun fill(path: Path2D, fillRule: CanvasFillRule = definedExternally)\n   fun stroke()\n   fun stroke(path: Path2D)\n   fun clip(fillRule: CanvasFillRule = definedExternally)\n   fun clip(path: Path2D, fillRule: CanvasFillRule = definedExternally)\n   fun resetClip()\n   fun isPointInPath(x: Double, y: Double, fillRule: CanvasFillRule = definedExternally): Boolean\n   fun isPointInPath(path: Path2D, x: Double, y: Double, fillRule: CanvasFillRule = definedExternally): Boolean\n   fun isPointInStroke(x: Double, y: Double): Boolean\n   fun isPointInStroke(path: Path2D, x: Double, y: Double): Boolean\n}\n\npublic external interface CanvasUserInterface {\n   fun drawFocusIfNeeded(element: Element)\n   fun drawFocusIfNeeded(path:

Path2D, element: Element)\n   fun scrollPathIntoView()\n   fun scrollPathIntoView(path: Path2D)\n}\n\npublic external interface CanvasText {\n   fun fillText(text: String, x: Double, y: Double, maxWidth: Double = definedExternally)\n   fun strokeText(text: String, x: Double, y: Double, maxWidth: Double = definedExternally)\n   fun measureText(text: String): TextMetrics\n}\n\npublic external interface CanvasDrawImage {\n   fun drawImage(image: CanvasImageSource, dx: Double, dy: Double)\n   fun drawImage(image: CanvasImageSource, dx: Double, dy: Double, dw: Double, dh: Double)\n   fun drawImage(image: CanvasImageSource, sx: Double, sy: Double, sw: Double, sh: Double, dx: Double, dy: Double, dw: Double, dh: Double)\n}\n\npublic external interface CanvasHitRegion {\n   fun addHitRegion(options: HitRegionOptions = definedExternally)\n   fun removeHitRegion(id: String)\n   fun clearHitRegions()\n}\n\npublic external interface CanvasImageData {\n   fun createImageData(sw: Double, sh: Double): ImageData\n   fun createImageData(imagedata: ImageData): ImageData\n   fun getImageData(sx: Double, sy: Double, sw: Double, sh: Double): ImageData\n   fun putImageData(imagedata: ImageData, dx: Double, dy: Double)\n   fun putImageData(imagedata: ImageData, dx: Double, dy: Double, dirtyX: Double, dirtyY: Double, dirtyWidth: Double, dirtyHeight: Double)\n}\n\npublic external interface CanvasPathDrawingStyles {\n   var lineWidth: Double\n   var lineCap: CanvasLineCap\n   var lineJoin: CanvasLineJoin\n   var miterLimit: Double\n   var lineDashOffset: Double\n   fun setLineDash(segments: Array<Double>)\n   fun getLineDash(): Array<Double>\n}\n\npublic external interface CanvasTextDrawingStyles {\n   var font: String\n   var textAlign: CanvasTextAlign\n   var textBaseline: CanvasTextBaseline\n   var direction: CanvasDirection\n}\n\npublic external interface CanvasPath {\n   fun closePath()\n   fun moveTo(x: Double, y: Double)\n   fun lineTo(x: Double, y: Double)\n   fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)\n   fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)\n   fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)\n   fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)\n   fun rect(x: Double, y: Double, w: Double, h: Double)\n   fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean = definedExternally)\n   fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean = definedExternally)\n}\n\n/**\n * Exposes the JavaScript [CanvasGradient](https://developer.mozilla.org/en/docs/Web/API/CanvasGradient) to Kotlin\n */\npublic external abstract class CanvasGradient {\n   fun addColorStop(offset: Double, color: String)\n}\n\n/**\n * Exposes the JavaScript [CanvasPattern](https://developer.mozilla.org/en/docs/Web/API/CanvasPattern) to Kotlin\n */\npublic external abstract class CanvasPattern {\n   fun setTransform(transform: dynamic = definedExternally)\n}\n\n/**\n * Exposes the JavaScript [TextMetrics](https://developer.mozilla.org/en/docs/Web/API/TextMetrics) to Kotlin\n */\npublic external abstract class TextMetrics {\n   open val width: Double\n   open val actualBoundingBoxLeft: Double\n   open val actualBoundingBoxRight: Double\n   open val fontBoundingBoxAscent: Double\n   open val fontBoundingBoxDescent: Double\n   open val actualBoundingBoxAscent: Double\n   open val actualBoundingBoxDescent: Double\n   open val emHeightAscent: Double\n   open val emHeightDescent: Double\n   open val hangingBaseline: Double\n   open val alphabeticBaseline: Double\n   open val ideographicBaseline: Double\n}\n\npublic external interface HitRegionOptions {\n   var path: Path2D? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n   var fillRule: CanvasFillRule? /* = CanvasFillRule.NONZERO */\n   get() = definedExternally\n   set(value) = definedExternally\n   var id: String? /* = \"\" */\n   get() = definedExternally\n   set(value) = definedExternally\n   var parentID: String? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n   var cursor: String? /* = \"inherit\" */\n   get() = definedExternally\n   set(value) = definedExternally\n   var control: Element? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n   var label: String? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n   var role: String? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun HitRegionOptions(path: Path2D? = null, fillRule: CanvasFillRule? = CanvasFillRule.NONZERO, id: String? = \"\", parentID: String? = null, cursor: String? = \"inherit\", control: Element? = null, label: String? = null, role: String? = null): HitRegionOptions {\n   val

o = js(\"({})\")\n   o[\"path\"] = path\n   o[\"fillRule\"] = fillRule\n   o[\"id\"] = id\n   o[\"parentID\"] = parentID\n   o[\"cursor\"] = cursor\n   o[\"control\"] = control\n   o[\"label\"] = label\n   o[\"role\"] = role\n   return o\n}\n\n/**\n * Exposes the JavaScript [ImageData](https://developer.mozilla.org/en/docs/Web/API/ImageData) to Kotlin\n */\npublic external open class ImageData : ImageBitmapSource, TexImageSource {\n   constructor(sw: Int, sh: Int)\n   constructor(data: Uint8ClampedArray, sw: Int, sh: Int = definedExternally)\n   open val width: Int\n   open val height: Int\n   open val data: Uint8ClampedArray\n}\n\n/**\n * Exposes the JavaScript [Path2D](https://developer.mozilla.org/en/docs/Web/API/Path2D) to Kotlin\n */\npublic external open class Path2D() : CanvasPath {\n   constructor(path: Path2D)\n   constructor(paths: Array<Path2D>, fillRule: CanvasFillRule = definedExternally)\n   constructor(d: String)\n   fun addPath(path: Path2D, transform: dynamic = definedExternally)\n   override fun closePath()\n   override fun moveTo(x: Double, y: Double)\n   override fun lineTo(x: Double, y: Double)\n   override fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)\n   override fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)\n   override fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)\n   override fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)\n   override fun rect(x: Double, y: Double, w: Double, h: Double)\n   override fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean /* = definedExternally */)\n   override fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean /* = definedExternally */)\n}\n\n/**\n * Exposes the JavaScript [ImageBitmapRenderingContext](https://developer.mozilla.org/en/docs/Web/API/ImageBitmapRenderingContext) to Kotlin\n */\npublic external abstract class ImageBitmapRenderingContext {\n   open val canvas: HTMLCanvasElement\n   fun transferFromImageBitmap(bitmap: ImageBitmap?)\n}\n\npublic external interface ImageBitmapRenderingContextSettings {\n   var alpha: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ImageBitmapRenderingContextSettings(alpha: Boolean? = true): ImageBitmapRenderingContextSettings {\n   val o = js(\"({})\")\n   o[\"alpha\"] = alpha\n   return o\n}\n\n/**\n * Exposes the JavaScript [CustomElementRegistry](https://developer.mozilla.org/en/docs/Web/API/CustomElementRegistry) to Kotlin\n */\npublic external abstract class CustomElementRegistry {\n   fun define(name: String, constructor: () -> dynamic, options: ElementDefinitionOptions = definedExternally)\n   fun get(name: String): Any?\n   fun whenDefined(name: String): Promise<Unit>\n}\n\npublic external interface ElementDefinitionOptions {\n   var extends: String?\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ElementDefinitionOptions(extends: String? = undefined): ElementDefinitionOptions {\n   val o = js(\"({})\")\n   o[\"extends\"] = extends\n   return o\n}\n\npublic external interface ElementContentEditable {\n   var contentEditable: String\n   val isContentEditable: Boolean\n}\n\n/**\n * Exposes the JavaScript [DataTransfer](https://developer.mozilla.org/en/docs/Web/API/DataTransfer) to Kotlin\n */\npublic external abstract class DataTransfer {\n   open var dropEffect: String\n   open var effectAllowed: String\n   open val items: DataTransferItemList\n   open val types: Array<out String>\n   open val files: FileList\n   fun setDragImage(image: Element, x: Int, y: Int)\n   fun getData(format: String): String\n   fun setData(format: String, data: String)\n   fun clearData(format: String = definedExternally)\n}\n\n/**\n * Exposes the JavaScript [DataTransferItemList](https://developer.mozilla.org/en/docs/Web/API/DataTransferItemList) to Kotlin\n */\npublic external abstract class DataTransferItemList {\n   open val length: Int\n   fun add(data: String, type: String): DataTransferItem?\n   fun add(data: File): DataTransferItem?\n   fun remove(index: Int)\n   fun clear()\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun DataTransferItemList.get(index: Int): DataTransferItem? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript [DataTransferItem](https://developer.mozilla.org/en/docs/Web/API/DataTransferItem) to Kotlin\n */\npublic

```
external abstract class DataTransferItem {\n    open val kind: String\n    open val type: String\n    fun getAsString(_callback: ((String) -> Unit)?)\n    fun getAsFile(): File?\n}\n\n/**\n * Exposes the JavaScript [DragEvent](https://developer.mozilla.org/en/docs/Web/API/DragEvent) to Kotlin\n */\npublic external open class DragEvent(type: String, eventInitDict: DragEventInit = definedExternally) : MouseEvent {\n    open val dataTransfer: DataTransfer?\n\n    companion object {\n        val NONE: Short\n        val CAPTURING_PHASE: Short\n        val AT_TARGET: Short\n        val BUBBLING_PHASE: Short\n    }\n}\n\npublic external interface DragEventInit : MouseEventInit {\n    var dataTransfer: DataTransfer? /* = null */\n        get() = definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun DragEventInit(dataTransfer: DataTransfer? = null, screenX: Int? = 0, screenY: Int? = 0, clientX: Int? = 0, clientY: Int? = 0, button: Short? = 0, buttons: Short? = 0, relatedTarget: EventTarget? = null, region: String? = null, ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false, metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn: Boolean? = false, modifierFnLock: Boolean? = false, modifierHyper: Boolean? = false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false, modifierSymbol: Boolean? = false, modifierSymbolLock: Boolean? = false, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): DragEventInit {\n    val o = js(\"({})\")\n    o[\"dataTransfer\"] = dataTransfer\n    o[\"screenX\"] = screenX\n    o[\"screenY\"] = screenY\n    o[\"clientX\"] = clientX\n    o[\"clientY\"] = clientY\n    o[\"button\"] = button\n    o[\"buttons\"] = buttons\n    o[\"relatedTarget\"] = relatedTarget\n    o[\"region\"] = region\n    o[\"ctrlKey\"] = ctrlKey\n    o[\"shiftKey\"] = shiftKey\n    o[\"altKey\"] = altKey\n    o[\"metaKey\"] = metaKey\n    o[\"modifierAltGraph\"] = modifierAltGraph\n    o[\"modifierCapsLock\"] = modifierCapsLock\n    o[\"modifierFn\"] = modifierFn\n    o[\"modifierFnLock\"] = modifierFnLock\n    o[\"modifierHyper\"] = modifierHyper\n    o[\"modifierNumLock\"] = modifierNumLock\n    o[\"modifierScrollLock\"] = modifierScrollLock\n    o[\"modifierSuper\"] = modifierSuper\n    o[\"modifierSymbol\"] = modifierSymbol\n    o[\"modifierSymbolLock\"] = modifierSymbolLock\n    o[\"view\"] = view\n    o[\"detail\"] = detail\n    o[\"bubbles\"] = bubbles\n    o[\"cancelable\"] = cancelable\n    o[\"composed\"] = composed\n    return o\n}\n\n/**\n * Exposes the JavaScript [Window](https://developer.mozilla.org/en/docs/Web/API/Window) to Kotlin\n */\npublic external abstract class Window : EventTarget, GlobalEventHandlers, WindowEventHandlers, WindowOrWorkerGlobalScope, WindowSessionStorage, WindowLocalStorage, GlobalPerformance, UnionMessagePortOrWindowProxy {\n    open val window: Window\n    open val self: Window\n    open val document: Document\n    open var name: String\n    open val location: Location\n    open val history: History\n    open val customElements: CustomElementRegistry\n    open val locationbar: BarProp\n    open val menubar: BarProp\n    open val personalbar: BarProp\n    open val scrollbars: BarProp\n    open val statusbar: BarProp\n    open val toolbar: BarProp\n    open var status: String\n    open val closed: Boolean\n    open val frames: Window\n    open val length: Int\n    open val top: Window\n    open var opener: Any?\n    open val parent: Window\n    open val frameElement: Element?\n    open val navigator: Navigator\n    open val applicationCache: ApplicationCache\n    open val external: External\n    open val screen: Screen\n    open val innerWidth: Int\n    open val innerHeight: Int\n    open val scrollX: Double\n    open val pageXOffset: Double\n    open val scrollY: Double\n    open val pageYOffset: Double\n    open val screenX: Int\n    open val screenY: Int\n    open val outerWidth: Int\n    open val outerHeight: Int\n    open val devicePixelRatio: Double\n    fun close()\n    fun stop()\n    fun focus()\n    fun blur()\n    fun open(url: String = definedExternally, target: String = definedExternally, features: String = definedExternally): Window?\n    fun alert()\n    fun alert(message: String)\n    fun confirm(message: String = definedExternally): Boolean\n    fun prompt(message: String = definedExternally, default: String = definedExternally): String?\n    fun print()\n    fun requestAnimationFrame(callback: (Double) -> Unit): Int\n    fun cancelAnimationFrame(handle: Int)\n    fun postMessage(message: Any?, targetOrigin: String, transfer: Array<dynamic> = definedExternally)\n    fun captureEvents()\n    fun releaseEvents()\n    fun matchMedia(query: String): MediaQueryList\n    fun moveTo(x: Int, y: Int)\n    fun moveBy(x: Int, y: Int)\n    fun resizeTo(x: Int, y: Int)\n    fun resizeBy(x: Int, y: Int)\n    fun scroll(options: ScrollToOptions = definedExternally)\n    fun scroll(x: Double, y: Double)\n    fun scrollTo(options:
```

```
ScrollToOptions = definedExternally)\n    fun scrollTo(x: Double, y: Double)\n    fun scrollBy(options:
ScrollToOptions = definedExternally)\n    fun scrollBy(x: Double, y: Double)\n    fun getComputedStyle(elt:
Element, pseudoElt: String? = definedExternally):
CSSStyleDeclaration\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Window.get(name: String):
dynamic = asDynamic()[name]\n\npublic external abstract class BarProp {\n    open val visible: Boolean\n}\n\n/**\n
* Exposes the JavaScript [History](https://developer.mozilla.org/en/docs/Web/API/History) to Kotlin\n */\npublic
external abstract class History {\n    open val length: Int\n    open var scrollRestoration: ScrollRestoration\n    open
val state: Any?\n    fun go(delta: Int = definedExternally)\n    fun back()\n    fun forward()\n    fun pushState(data:
Any?, title: String, url: String? = definedExternally)\n    fun replaceState(data: Any?, title: String, url: String? =
definedExternally)\n}\n\n/**\n * Exposes the JavaScript
[Location](https://developer.mozilla.org/en/docs/Web/API/Location) to Kotlin\n */\npublic external abstract class
Location {\n    open var href: String\n    open val origin: String\n    open var protocol: String\n    open var host:
String\n    open var hostname: String\n    open var port: String\n    open var pathname: String\n    open var search:
String\n    open var hash: String\n    open val ancestorOrigins: Array<out String>\n    fun assign(url: String)\n    fun
replace(url: String)\n    fun reload()\n}\n\n/**\n * Exposes the JavaScript
[PopStateEvent](https://developer.mozilla.org/en/docs/Web/API/PopStateEvent) to Kotlin\n */\npublic external
open class PopStateEvent(type: String, eventInitDict: PopStateEventInit = definedExternally) : Event {\n    open val
state: Any?\n\n    companion object {\n        val NONE: Short\n        val CAPTURING_PHASE: Short\n        val
AT_TARGET: Short\n        val BUBBLING_PHASE: Short\n    }\n}\n\npublic external interface PopStateEventInit
: EventInit {\n    var state: Any? /* = null */\n        get() = definedExternally\n        set(value) =
definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun PopStateEventInit(state: Any? = null,
bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): PopStateEventInit {\n    val o
= js(\"({})\")\n    o[\"state\"] = state\n    o[\"bubbles\"] = bubbles\n    o[\"cancelable\"] = cancelable\n
o[\"composed\"] = composed\n    return o\n}\n\n/**\n * Exposes the JavaScript
[HashChangeEvent](https://developer.mozilla.org/en/docs/Web/API/HashChangeEvent) to Kotlin\n */\npublic
external open class HashChangeEvent(type: String, eventInitDict: HashChangeEventInit = definedExternally) :
Event {\n    open val oldURL: String\n    open val newURL: String\n\n    companion object {\n        val NONE:
Short\n        val CAPTURING_PHASE: Short\n        val AT_TARGET: Short\n        val BUBBLING_PHASE:
Short\n    }\n}\n\npublic external interface HashChangeEventInit : EventInit {\n    var oldURL: String? /* = \"\" */\n
    get() = definedExternally\n        set(value) = definedExternally\n    var newURL: String? /* = \"\" */\n        get() =
definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun HashChangeEventInit(oldURL:
String? = \"\", newURL: String? = \"\", bubbles: Boolean? = false, cancelable: Boolean? = false, composed:
Boolean? = false): HashChangeEventInit {\n    val o = js(\"({})\")\n    o[\"oldURL\"] = oldURL\n    o[\"newURL\"]
= newURL\n    o[\"bubbles\"] = bubbles\n    o[\"cancelable\"] = cancelable\n    o[\"composed\"] = composed\n
return o\n}\n\n/**\n * Exposes the JavaScript
[PageTransitionEvent](https://developer.mozilla.org/en/docs/Web/API/PageTransitionEvent) to Kotlin\n */\npublic
external open class PageTransitionEvent(type: String, eventInitDict: PageTransitionEventInit = definedExternally) :
Event {\n    open val persisted: Boolean\n\n    companion object {\n        val NONE: Short\n        val
CAPTURING_PHASE: Short\n        val AT_TARGET: Short\n        val BUBBLING_PHASE: Short\n
}\n}\n\npublic external interface PageTransitionEventInit : EventInit {\n    var persisted: Boolean? /* = false */\n
get() = definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun PageTransitionEventInit(persisted:
Boolean? = false, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false):
PageTransitionEventInit {\n    val o = js(\"({})\")\n    o[\"persisted\"] = persisted\n    o[\"bubbles\"] = bubbles\n
o[\"cancelable\"] = cancelable\n    o[\"composed\"] = composed\n    return o\n}\n\n/**\n * Exposes the JavaScript
```

[BeforeUnloadEvent](https://developer.mozilla.org/en/docs/Web/API/BeforeUnloadEvent) to Kotlin\n */\npublic external open class BeforeUnloadEvent : Event {\n var returnValue: String\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external abstract class ApplicationCache : EventTarget {\n open val status: Short\n open var onchecking: ((Event) -> dynamic)?\n open var onerror: ((Event) -> dynamic)?\n open var onnoupdate: ((Event) -> dynamic)?\n open var ondownloading: ((Event) -> dynamic)?\n open var onprogress: ((ProgressEvent) -> dynamic)?\n open var onupdateready: ((Event) -> dynamic)?\n open var oncached: ((Event) -> dynamic)?\n open var onobsolete: ((Event) -> dynamic)?\n fun update()\n fun abort()\n fun swapCache()\n\n companion object {\n val UNCACHED: Short\n val IDLE: Short\n val CHECKING: Short\n val DOWNLOADING: Short\n val UPDATEREADY: Short\n val OBSOLETE: Short\n }\n}\n\n/**\n * Exposes the JavaScript [NavigatorOnLine](https://developer.mozilla.org/en/docs/Web/API/NavigatorOnLine) to Kotlin\n */\npublic external interface NavigatorOnLine {\n val onLine: Boolean\n}\n\n/**\n * Exposes the JavaScript [ErrorEvent](https://developer.mozilla.org/en/docs/Web/API/ErrorEvent) to Kotlin\n */\npublic external open class ErrorEvent(type: String, eventInitDict: ErrorEventInit = definedExternally) : Event {\n open val message: String\n open val filename: String\n open val lineno: Int\n open val colno: Int\n open val error: Any?\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface ErrorEventInit : EventInit {\n var message: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n var filename: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n var lineno: Int? /* = 0 */\n get() = definedExternally\n set(value) = definedExternally\n var colno: Int? /* = 0 */\n get() = definedExternally\n set(value) = definedExternally\n var error: Any? /* = null */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ErrorEventInit(message: String? = \"\", filename: String? = \"\", lineno: Int? = 0, colno: Int? = 0, error: Any? = null, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): ErrorEventInit {\n val o = js(\"({})\")\n o[\"message\"] = message\n o[\"filename\"] = filename\n o[\"lineno\"] = lineno\n o[\"colno\"] = colno\n o[\"error\"] = error\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [PromiseRejectionEvent](https://developer.mozilla.org/en/docs/Web/API/PromiseRejectionEvent) to Kotlin\n */\npublic external open class PromiseRejectionEvent(type: String, eventInitDict: PromiseRejectionEventInit) : Event {\n open val promise: Promise<Any?>\n open val reason: Any?\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface PromiseRejectionEventInit : EventInit {\n var promise: Promise<Any?>?\n var reason: Any?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun PromiseRejectionEventInit(promise: Promise<Any?>?, reason: Any? = undefined, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): PromiseRejectionEventInit {\n val o = js(\"({})\")\n o[\"promise\"] = promise\n o[\"reason\"] = reason\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [GlobalEventHandlers](https://developer.mozilla.org/en/docs/Web/API/GlobalEventHandlers) to Kotlin\n */\npublic external interface GlobalEventHandlers {\n var onabort: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onblur: ((FocusEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var oncancel: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var oncanplay: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var oncanplaythrough: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onchange: ((Event) -> dynamic)?\n get() =

definedExternally\n set(value) = definedExternally\n var onclick: ((MouseEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onclose: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var oncontextmenu: ((MouseEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var oncuechange: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var ondblclick: ((MouseEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var ondrag: ((DragEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var ondragend: ((DragEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var ondragenter: ((DragEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var ondragexit: ((DragEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var ondragleave: ((DragEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var ondragover: ((DragEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var ondragstart: ((DragEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var ondrop: ((DragEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var ondurationchange: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onemptied: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onended: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onfocus: ((FocusEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var oninput: ((InputEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var oninvalid: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onkeydown: ((KeyboardEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onkeypress: ((KeyboardEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onkeyup: ((KeyboardEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onload: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onloadeddata: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onloadedmetadata: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onloadend: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onloadstart: ((ProgressEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onmousedown: ((MouseEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onmouseenter: ((MouseEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onmouseleave: ((MouseEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onmousemove: ((MouseEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onmouseout: ((MouseEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onmouseover: ((MouseEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onmouseup: ((MouseEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onwheel: ((WheelEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onpause: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onplay: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onplaying: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onprogress: ((ProgressEvent) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onratechange: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onreset: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onresize: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onscroll: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onseeked: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n var onseeking: ((Event) -> dynamic)?\n get() = definedExternally\n

```
set(value) = definedExternally\n    var onselect: ((Event) -> dynamic)?\n        get() = definedExternally\n
set(value) = definedExternally\n    var onshow: ((Event) -> dynamic)?\n        get() = definedExternally\n
set(value) = definedExternally\n    var onstalled: ((Event) -> dynamic)?\n        get() = definedExternally\n
set(value) = definedExternally\n    var onsubmit: ((Event) -> dynamic)?\n        get() = definedExternally\n
set(value) = definedExternally\n    var onsuspend: ((Event) -> dynamic)?\n        get() = definedExternally\n
set(value) = definedExternally\n    var ontimeupdate: ((Event) -> dynamic)?\n        get() = definedExternally\n
set(value) = definedExternally\n    var ontoggle: ((Event) -> dynamic)?\n        get() = definedExternally\n
set(value) = definedExternally\n    var onvolumechange: ((Event) -> dynamic)?\n        get() = definedExternally\n
 set(value) = definedExternally\n    var onwaiting: ((Event) -> dynamic)?\n        get() = definedExternally\n
set(value) = definedExternally\n    var ongotpointercapture: ((PointerEvent) -> dynamic)?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var onlostpointercapture: ((PointerEvent) -> dynamic)?\n
    get() = definedExternally\n        set(value) = definedExternally\n    var onpointerdown: ((PointerEvent) ->
dynamic)?\n        get() = definedExternally\n        set(value) = definedExternally\n    var onpointermove:
((PointerEvent) -> dynamic)?\n        get() = definedExternally\n        set(value) = definedExternally\n    var
onpointerup: ((PointerEvent) -> dynamic)?\n        get() = definedExternally\n        set(value) = definedExternally\n
var onpointercancel: ((PointerEvent) -> dynamic)?\n        get() = definedExternally\n        set(value) =
definedExternally\n    var onpointerover: ((PointerEvent) -> dynamic)?\n        get() = definedExternally\n
set(value) = definedExternally\n    var onpointerout: ((PointerEvent) -> dynamic)?\n        get() = definedExternally\n
    set(value) = definedExternally\n    var onpointerenter: ((PointerEvent) -> dynamic)?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var onpointerleave: ((PointerEvent) -> dynamic)?\n
get() = definedExternally\n        set(value) = definedExternally\n}\n\n/**\n * Exposes the JavaScript
[WindowEventHandlers](https://developer.mozilla.org/en/docs/Web/API/WindowEventHandlers) to Kotlin\n
*/\npublic external interface WindowEventHandlers {\n    var onafterprint: ((Event) -> dynamic)?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var onbeforeprint: ((Event) -> dynamic)?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var onbeforeunload: ((BeforeUnloadEvent) ->
String?)?\n        get() = definedExternally\n        set(value) = definedExternally\n    var onhashchange:
((HashChangeEvent) -> dynamic)?\n        get() = definedExternally\n        set(value) = definedExternally\n    var
onlanguagechange: ((Event) -> dynamic)?\n        get() = definedExternally\n        set(value) = definedExternally\n
var onmessage: ((MessageEvent) -> dynamic)?\n        get() = definedExternally\n        set(value) =
definedExternally\n    var onoffline: ((Event) -> dynamic)?\n        get() = definedExternally\n        set(value) =
definedExternally\n    var ononline: ((Event) -> dynamic)?\n        get() = definedExternally\n        set(value) =
definedExternally\n    var onpagehide: ((PageTransitionEvent) -> dynamic)?\n        get() = definedExternally\n
set(value) = definedExternally\n    var onpageshow: ((PageTransitionEvent) -> dynamic)?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var onpopstate: ((PopStateEvent) -> dynamic)?\n
get() = definedExternally\n        set(value) = definedExternally\n    var onrejectionhandled: ((Event) -> dynamic)?\n
   get() = definedExternally\n        set(value) = definedExternally\n    var onstorage: ((StorageEvent) -> dynamic)?\n
    get() = definedExternally\n        set(value) = definedExternally\n    var onunhandledrejection:
((PromiseRejectionEvent) -> dynamic)?\n        get() = definedExternally\n        set(value) = definedExternally\n
var onunload: ((Event) -> dynamic)?\n        get() = definedExternally\n        set(value) =
definedExternally\n}\n\npublic external interface DocumentAndElementEventHandlers {\n    var oncopy:
((ClipboardEvent) -> dynamic)?\n        get() = definedExternally\n        set(value) = definedExternally\n    var oncut:
((ClipboardEvent) -> dynamic)?\n        get() = definedExternally\n        set(value) = definedExternally\n    var
onpaste: ((ClipboardEvent) -> dynamic)?\n        get() = definedExternally\n        set(value) =
definedExternally\n}\n\n/**\n * Exposes the JavaScript
[WindowOrWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/WindowOrWorkerGlobalScope)
to Kotlin\n */\npublic external interface WindowOrWorkerGlobalScope {\n    val origin: String\n    val caches:
CacheStorage\n    fun btoa(data: String): String\n    fun atob(data: String): String\n    fun setTimeout(handler:
dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int\n    fun clearTimeout(handle: Int =
```

definedExternally)\n   fun setInterval(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int\n   fun clearInterval(handle: Int = definedExternally)\n   fun createImageBitmap(image: ImageBitmapSource, options: ImageBitmapOptions = definedExternally): Promise<ImageBitmap>\n   fun createImageBitmap(image: ImageBitmapSource, sx: Int, sy: Int, sw: Int, sh: Int, options: ImageBitmapOptions = definedExternally): Promise<ImageBitmap>\n   fun fetch(input: dynamic, init: RequestInit = definedExternally): Promise<Response>\n}\n\n/**\n * Exposes the JavaScript [Navigator](https://developer.mozilla.org/en/docs/Web/API/Navigator) to Kotlin\n */\npublic external abstract class Navigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorContentUtils, NavigatorCookies, NavigatorPlugins, NavigatorConcurrentHardware {\n   open val clipboard: Clipboard\n   open val mediaDevices: MediaDevices\n   open val maxTouchPoints: Int\n   open val serviceWorker: ServiceWorkerContainer\n   fun requestMediaKeySystemAccess(keySystem: String, supportedConfigurations: Array<MediaKeySystemConfiguration>): Promise<MediaKeySystemAccess>\n   fun getUserMedia(constraints: MediaStreamConstraints, successCallback: (MediaStream) -> Unit, errorCallback: (dynamic) -> Unit)\n   fun vibrate(pattern: dynamic): Boolean\n}\n\n/**\n * Exposes the JavaScript [NavigatorID](https://developer.mozilla.org/en/docs/Web/API/NavigatorID) to Kotlin\n */\npublic external interface NavigatorID {\n   val appCodeName: String\n   val appName: String\n   val appVersion: String\n   val platform: String\n   val product: String\n   val productSub: String\n   val userAgent: String\n   val vendor: String\n   val vendorSub: String\n   val oscpu: String\n   fun taintEnabled(): Boolean\n}\n\n/**\n * Exposes the JavaScript [NavigatorLanguage](https://developer.mozilla.org/en/docs/Web/API/NavigatorLanguage) to Kotlin\n */\npublic external interface NavigatorLanguage {\n   val language: String\n   val languages: Array<out String>\n}\n\npublic external interface NavigatorContentUtils {\n   fun registerProtocolHandler(scheme: String, url: String, title: String)\n   fun registerContentHandler(mimeType: String, url: String, title: String)\n   fun isProtocolHandlerRegistered(scheme: String, url: String): String\n   fun isContentHandlerRegistered(mimeType: String, url: String): String\n   fun unregisterProtocolHandler(scheme: String, url: String)\n   fun unregisterContentHandler(mimeType: String, url: String)\n}\n\npublic external interface NavigatorCookies {\n   val cookieEnabled: Boolean\n}\n\n/**\n * Exposes the JavaScript [NavigatorPlugins](https://developer.mozilla.org/en/docs/Web/API/NavigatorPlugins) to Kotlin\n */\npublic external interface NavigatorPlugins {\n   val plugins: PluginArray\n   val mimeTypes: MimeTypeArray\n   fun javaEnabled(): Boolean\n}\n\n/**\n * Exposes the JavaScript [PluginArray](https://developer.mozilla.org/en/docs/Web/API/PluginArray) to Kotlin\n */\npublic external abstract class PluginArray : ItemArrayLike<Plugin> {\n   fun refresh(reload: Boolean = definedExternally)\n   override fun item(index: Int): Plugin?\n   fun namedItem(name: String): Plugin?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun PluginArray.get(index: Int): Plugin? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun PluginArray.get(name: String): Plugin? = asDynamic()[name]\n\n/**\n * Exposes the JavaScript [MimeTypeArray](https://developer.mozilla.org/en/docs/Web/API/MimeTypeArray) to Kotlin\n */\npublic external abstract class MimeTypeArray : ItemArrayLike<MimeType> {\n   override fun item(index: Int): MimeType?\n   fun namedItem(name: String): MimeType?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun MimeTypeArray.get(index: Int): MimeType? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun MimeTypeArray.get(name: String): MimeType? = asDynamic()[name]\n\n/**\n * Exposes the JavaScript [Plugin](https://developer.mozilla.org/en/docs/Web/API/Plugin) to Kotlin\n */\npublic external abstract class Plugin : ItemArrayLike<MimeType> {\n   open val name: String\n   open val description: String\n   open val filename: String\n   override fun item(index: Int): MimeType?\n   fun namedItem(name: String): MimeType?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",

\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Plugin.get(index: Int): MimeType? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Plugin.get(name: String): MimeType? = asDynamic()[name]\n\n/**\n * Exposes the JavaScript [MimeType](https://developer.mozilla.org/en/docs/Web/API/MimeType) to Kotlin\n */\npublic external abstract class MimeType {\n open val type: String\n open val description: String\n open val suffixes: String\n open val enabledPlugin: Plugin\n}\n\n/**\n * Exposes the JavaScript [ImageBitmap](https://developer.mozilla.org/en/docs/Web/API/ImageBitmap) to Kotlin\n */\npublic external abstract class ImageBitmap : CanvasImageSource, TexImageSource {\n open val width: Int\n open val height: Int\n fun close()\n}\n\npublic external interface ImageBitmapOptions {\n var imageOrientation: ImageOrientation? /* = ImageOrientation.NONE */\n get() = definedExternally\n set(value) = definedExternally\n var premultiplyAlpha: PremultiplyAlpha? /* = PremultiplyAlpha.DEFAULT */\n get() = definedExternally\n set(value) = definedExternally\n var colorSpaceConversion: ColorSpaceConversion? /* = ColorSpaceConversion.DEFAULT */\n get() = definedExternally\n set(value) = definedExternally\n var resizeWidth: Int?\n get() = definedExternally\n set(value) = definedExternally\n var resizeHeight: Int?\n get() = definedExternally\n set(value) = definedExternally\n var resizeQuality: ResizeQuality? /* = ResizeQuality.LOW */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ImageBitmapOptions(imageOrientation: ImageOrientation? = ImageOrientation.NONE, premultiplyAlpha: PremultiplyAlpha? = PremultiplyAlpha.DEFAULT, colorSpaceConversion: ColorSpaceConversion? = ColorSpaceConversion.DEFAULT, resizeWidth: Int? = undefined, resizeHeight: Int? = undefined, resizeQuality: ResizeQuality? = ResizeQuality.LOW): ImageBitmapOptions {\n val o = js(\"({})\")\n o[\"imageOrientation\"] = imageOrientation\n o[\"premultiplyAlpha\"] = premultiplyAlpha\n o[\"colorSpaceConversion\"] = colorSpaceConversion\n o[\"resizeWidth\"] = resizeWidth\n o[\"resizeHeight\"] = resizeHeight\n o[\"resizeQuality\"] = resizeQuality\n return o\n}\n\n/**\n * Exposes the JavaScript [MessageEvent](https://developer.mozilla.org/en/docs/Web/API/MessageEvent) to Kotlin\n */\npublic external open class MessageEvent(type: String, eventInitDict: MessageEventInit = definedExternally) : Event {\n open val data: Any?\n open val origin: String\n open val lastEventId: String\n open val source: UnionMessagePortOrWindowProxy?\n open val ports: Array<out MessagePort>\n fun initMessageEvent(type: String, bubbles: Boolean, cancelable: Boolean, data: Any?, origin: String, lastEventId: String, source: UnionMessagePortOrWindowProxy?, ports: Array<MessagePort>)\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface MessageEventInit : EventInit {\n var data: Any? /* = null */\n get() = definedExternally\n set(value) = definedExternally\n var origin: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n var lastEventId: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n var source: UnionMessagePortOrWindowProxy? /* = null */\n get() = definedExternally\n set(value) = definedExternally\n var ports: Array<MessagePort>? /* = arrayOf() */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MessageEventInit(data: Any? = null, origin: String? = \"\", lastEventId: String? = \"\", source: UnionMessagePortOrWindowProxy? = null, ports: Array<MessagePort>? = arrayOf(), bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): MessageEventInit {\n val o = js(\"({})\")\n o[\"data\"] = data\n o[\"origin\"] = origin\n o[\"lastEventId\"] = lastEventId\n o[\"source\"] = source\n o[\"ports\"] = ports\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [EventSource](https://developer.mozilla.org/en/docs/Web/API/EventSource) to Kotlin\n */\npublic external open class EventSource(url: String, eventSourceInitDict: EventSourceInit = definedExternally) : EventTarget {\n open

```
val url: String\n    open val withCredentials: Boolean\n    open val readyState: Short\n    var onopen: ((Event) -> dynamic)?\n    var onmessage: ((MessageEvent) -> dynamic)?\n    var onerror: ((Event) -> dynamic)?\n    fun close()\n\n    companion object {\n        val CONNECTING: Short\n        val OPEN: Short\n        val CLOSED: Short\n    }\n}\n\npublic external interface EventSourceInit {\n    var withCredentials: Boolean? /* = false */\n        get() = definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun EventSourceInit(withCredentials: Boolean? = false): EventSourceInit {\n    val o = js(\"({})\")\n    o[\"withCredentials\"] = withCredentials\n    return o\n}\n\n/**\n * Exposes the JavaScript [WebSocket](https://developer.mozilla.org/en/docs/Web/API/WebSocket) to Kotlin\n */\npublic external open class WebSocket(url: String, protocols: dynamic = definedExternally) : EventTarget {\n    open val url: String\n    open val readyState: Short\n    open val bufferedAmount: Number\n    var onopen: ((Event) -> dynamic)?\n    var onerror: ((Event) -> dynamic)?\n    var onclose: ((Event) -> dynamic)?\n    open val extensions: String\n    open val protocol: String\n    var onmessage: ((MessageEvent) -> dynamic)?\n    var binaryType: BinaryType\n    fun close(code: Short = definedExternally, reason: String = definedExternally)\n    fun send(data: String)\n    fun send(data: Blob)\n    fun send(data: ArrayBuffer)\n    fun send(data: ArrayBufferView)\n\n    companion object {\n        val CONNECTING: Short\n        val OPEN: Short\n        val CLOSING: Short\n        val CLOSED: Short\n    }\n}\n\n/**\n * Exposes the JavaScript [CloseEvent](https://developer.mozilla.org/en/docs/Web/API/CloseEvent) to Kotlin\n */\npublic external open class CloseEvent(type: String, eventInitDict: CloseEventInit = definedExternally) : Event {\n    open val wasClean: Boolean\n    open val code: Short\n    open val reason: String\n\n    companion object {\n        val NONE: Short\n        val CAPTURING_PHASE: Short\n        val AT_TARGET: Short\n        val BUBBLING_PHASE: Short\n    }\n}\n\npublic external interface CloseEventInit : EventInit {\n    var wasClean: Boolean? /* = false */\n        get() = definedExternally\n        set(value) = definedExternally\n    var code: Short? /* = 0 */\n        get() = definedExternally\n        set(value) = definedExternally\n    var reason: String? /* = \"\" */\n        get() = definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun CloseEventInit(wasClean: Boolean? = false, code: Short? = 0, reason: String? = \"\", bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): CloseEventInit {\n    val o = js(\"({})\")\n    o[\"wasClean\"] = wasClean\n    o[\"code\"] = code\n    o[\"reason\"] = reason\n    o[\"bubbles\"] = bubbles\n    o[\"cancelable\"] = cancelable\n    o[\"composed\"] = composed\n    return o\n}\n\n/**\n * Exposes the JavaScript [MessageChannel](https://developer.mozilla.org/en/docs/Web/API/MessageChannel) to Kotlin\n */\npublic external open class MessageChannel {\n    open val port1: MessagePort\n    open val port2: MessagePort\n}\n\n/**\n * Exposes the JavaScript [MessagePort](https://developer.mozilla.org/en/docs/Web/API/MessagePort) to Kotlin\n */\npublic external abstract class MessagePort : EventTarget, UnionMessagePortOrWindowProxy, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {\n    open var onmessage: ((MessageEvent) -> dynamic)?\n    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)\n    fun start()\n    fun close()\n}\n\n/**\n * Exposes the JavaScript [BroadcastChannel](https://developer.mozilla.org/en/docs/Web/API/BroadcastChannel) to Kotlin\n */\npublic external open class BroadcastChannel(name: String) : EventTarget {\n    open val name: String\n    var onmessage: ((MessageEvent) -> dynamic)?\n    fun postMessage(message: Any?)\n    fun close()\n}\n\n/**\n * Exposes the JavaScript [WorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/WorkerGlobalScope) to Kotlin\n */\npublic external abstract class WorkerGlobalScope : EventTarget, WindowOrWorkerGlobalScope, GlobalPerformance {\n    open val self: WorkerGlobalScope\n    open val location: WorkerLocation\n    open val navigator: WorkerNavigator\n    open var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?\n    open var onlanguagechange: ((Event) -> dynamic)?\n    open var onoffline: ((Event) -> dynamic)?\n    open var ononline: ((Event) -> dynamic)?\n    open var onrejectionhandled: ((Event) -> dynamic)?\n    open var onunhandledrejection: ((PromiseRejectionEvent) -> dynamic)?\n    fun importScripts(vararg urls: String)\n}\n\n/**\n * Exposes the JavaScript [DedicatedWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/DedicatedWorkerGlobalScope) to
```

Kotlin\n */\npublic external abstract class DedicatedWorkerGlobalScope : WorkerGlobalScope {\n   open var onmessage: ((MessageEvent) -> dynamic)?\n   fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)\n   fun close()\n}\n\n/**\n * Exposes the JavaScript [SharedWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/SharedWorkerGlobalScope) to Kotlin\n */\npublic external abstract class SharedWorkerGlobalScope : WorkerGlobalScope {\n   open val name: String\n   open val applicationCache: ApplicationCache\n   open var onconnect: ((Event) -> dynamic)?\n   fun close()\n}\n\n/**\n * Exposes the JavaScript [AbstractWorker](https://developer.mozilla.org/en/docs/Web/API/AbstractWorker) to Kotlin\n */\npublic external interface AbstractWorker {\n   var onerror: ((Event) -> dynamic)?\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n/**\n * Exposes the JavaScript [Worker](https://developer.mozilla.org/en/docs/Web/API/Worker) to Kotlin\n */\npublic external open class Worker(scriptURL: String, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {\n   var onmessage: ((MessageEvent) -> dynamic)?\n   override var onerror: ((Event) -> dynamic)?\n   fun terminate()\n   fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)\n}\n\npublic external interface WorkerOptions {\n   var type: WorkerType? /* = WorkerType.CLASSIC */\n   get() = definedExternally\n   set(value) = definedExternally\n   var credentials: RequestCredentials? /* = RequestCredentials.OMIT */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun WorkerOptions(type: WorkerType? = WorkerType.CLASSIC, credentials: RequestCredentials? = RequestCredentials.OMIT): WorkerOptions {\n   val o = js(\"({})\")\n   o[\"type\"] = type\n   o[\"credentials\"] = credentials\n   return o\n}\n\n/**\n * Exposes the JavaScript [SharedWorker](https://developer.mozilla.org/en/docs/Web/API/SharedWorker) to Kotlin\n */\npublic external open class SharedWorker(scriptURL: String, name: String = definedExternally, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {\n   open val port: MessagePort\n   override var onerror: ((Event) -> dynamic)?\n}\n\n/**\n * Exposes the JavaScript [NavigatorConcurrentHardware](https://developer.mozilla.org/en/docs/Web/API/NavigatorConcurrentHardware) to Kotlin\n */\npublic external interface NavigatorConcurrentHardware {\n   val hardwareConcurrency: Number\n}\n\n/**\n * Exposes the JavaScript [WorkerNavigator](https://developer.mozilla.org/en/docs/Web/API/WorkerNavigator) to Kotlin\n */\npublic external abstract class WorkerNavigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorConcurrentHardware {\n   open val serviceWorker: ServiceWorkerContainer\n}\n\n/**\n * Exposes the JavaScript [WorkerLocation](https://developer.mozilla.org/en/docs/Web/API/WorkerLocation) to Kotlin\n */\npublic external abstract class WorkerLocation {\n   open val href: String\n   open val origin: String\n   open val protocol: String\n   open val host: String\n   open val hostname: String\n   open val port: String\n   open val pathname: String\n   open val search: String\n   open val hash: String\n}\n\n/**\n * Exposes the JavaScript [Storage](https://developer.mozilla.org/en/docs/Web/API/Storage) to Kotlin\n */\npublic external abstract class Storage {\n   open val length: Int\n   fun key(index: Int): String?\n   fun removeItem(key: String)\n   fun clear()\n   fun getItem(key: String): String?\n   fun setItem(key: String, value: String)\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Storage.get(key: String): String? = asDynamic()[key]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Storage.set(key: String, value: String) { asDynamic()[key] = value }\n\n/**\n * Exposes the JavaScript [WindowSessionStorage](https://developer.mozilla.org/en/docs/Web/API/WindowSessionStorage) to Kotlin\n */\npublic external interface WindowSessionStorage {\n   val sessionStorage: Storage\n}\n\n/**\n * Exposes the JavaScript [WindowLocalStorage](https://developer.mozilla.org/en/docs/Web/API/WindowLocalStorage) to Kotlin\n */\npublic external interface WindowLocalStorage {\n   val localStorage: Storage\n}\n\n/**\n * Exposes the JavaScript [StorageEvent](https://developer.mozilla.org/en/docs/Web/API/StorageEvent) to Kotlin\n */\npublic external open class StorageEvent(type: String, eventInitDict: StorageEventInit = definedExternally) : Event {\n

```
open val key: String?\n    open val oldValue: String?\n    open val newValue: String?\n    open val url: String\n
open val storageArea: Storage?\n\n    companion object {\n        val NONE: Short\n        val CAPTURING_PHASE:
Short\n        val AT_TARGET: Short\n        val BUBBLING_PHASE: Short\n    }\n}\n\npublic external interface
StorageEventInit : EventInit {\n    var key: String? /* = null */\n        get() = definedExternally\n        set(value) =
definedExternally\n    var oldValue: String? /* = null */\n        get() = definedExternally\n        set(value) =
definedExternally\n    var newValue: String? /* = null */\n        get() = definedExternally\n        set(value) =
definedExternally\n    var url: String? /* = \"\" */\n        get() = definedExternally\n        set(value) =
definedExternally\n    var storageArea: Storage? /* = null */\n        get() = definedExternally\n        set(value) =
definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun StorageEventInit(key: String? = null,
oldValue: String? = null, newValue: String? = null, url: String? = \"\", storageArea: Storage? = null, bubbles:
Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): StorageEventInit {\n    val o =
js(\"({})\")\n    o[\"key\"] = key\n    o[\"oldValue\"] = oldValue\n    o[\"newValue\"] = newValue\n    o[\"url\"] =
url\n    o[\"storageArea\"] = storageArea\n    o[\"bubbles\"] = bubbles\n    o[\"cancelable\"] = cancelable\n
o[\"composed\"] = composed\n    return o\n}\n\npublic external abstract class HTMLAppletElement :
HTMLElement {\n    open var align: String\n    open var alt: String\n    open var archive: String\n    open var code:
String\n    open var codeBase: String\n    open var height: String\n    open var hspace: Int\n    open var name:
String\n    open var _object: String\n    open var vspace: Int\n    open var width: String\n\n    companion object {\n
val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val
CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE:
Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[HTMLMarqueeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMarqueeElement) to Kotlin\n
*/\npublic external abstract class HTMLMarqueeElement : HTMLElement {\n    open var behavior: String\n    open
var bgColor: String\n    open var direction: String\n    open var height: String\n    open var hspace: Int\n    open var
loop: Int\n    open var scrollAmount: Int\n    open var scrollDelay: Int\n    open var trueSpeed: Boolean\n    open var
vspace: Int\n    open var width: String\n    open var onbounce: ((Event) -> dynamic)?\n    open var onfinish: ((Event)
-> dynamic)?\n    open var onstart: ((Event) -> dynamic)?\n    fun start()\n    fun stop()\n\n    companion object {\n
val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val
CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE:
Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[HTMLFrameSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFrameSetElement) to Kotlin\n
*/\npublic external abstract class HTMLFrameSetElement : HTMLElement, WindowEventHandlers {\n    open var
cols: String\n    open var rows: String\n\n    companion object {\n        val ELEMENT_NODE: Short\n        val
ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val
ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE: Short\n        val
PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
```

DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external abstract class HTMLFrameElement : HTMLElement {\n open var name: String\n open var scrolling: String\n open var src: String\n open var frameBorder: String\n open var longDesc: String\n open var noResize: Boolean\n open val contentDocument: Document?\n open val contentWindow: Window?\n open var marginHeight: String\n open var marginWidth: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external abstract class HTMLDirectoryElement : HTMLElement {\n open var compact: Boolean\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [HTMLFontElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFontElement) to Kotlin\n */\npublic external abstract class HTMLFontElement : HTMLElement {\n open var color: String\n open var face: String\n open var size: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external interface External {\n fun AddSearchProvider()\n fun IsSearchProviderInstalled()\n}\n\npublic external interface EventInit {\n var bubbles: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n var cancelable: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n var composed: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun EventInit(bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): EventInit {\n val o = js(\"({})\")\n o[\"bubbles\"] =

bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\n/**\n * Exposes the JavaScript [CustomEvent](https://developer.mozilla.org/en/docs/Web/API/CustomEvent) to Kotlin\n */\npublic external open class CustomEvent(type: String, eventInitDict: CustomEventInit = definedExternally) : Event {\n   open val detail: Any?\n   fun initCustomEvent(type: String, bubbles: Boolean, cancelable: Boolean, detail: Any?)\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface CustomEventInit : EventInit {\n   var detail: Any? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun CustomEventInit(detail: Any? = null, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): CustomEventInit {\n   val o = js(\"({})\")\n   o[\"detail\"] = detail\n   o[\"bubbles\"] = bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\npublic external interface EventListenerOptions {\n   var capture: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun EventListenerOptions(capture: Boolean? = false): EventListenerOptions {\n   val o = js(\"({})\")\n   o[\"capture\"] = capture\n   return o\n}\n\npublic external interface AddEventListenerOptions : EventListenerOptions {\n   var passive: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var once: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun AddEventListenerOptions(passive: Boolean? = false, once: Boolean? = false, capture: Boolean? = false): AddEventListenerOptions {\n   val o = js(\"({})\")\n   o[\"passive\"] = passive\n   o[\"once\"] = once\n   o[\"capture\"] = capture\n   return o\n}\n\npublic external interface NonElementParentNode {\n   fun getElementById(elementId: String): Element?\n}\n\n/**\n * Exposes the JavaScript [DocumentOrShadowRoot](https://developer.mozilla.org/en/docs/Web/API/DocumentOrShadowRoot) to Kotlin\n */\npublic external interface DocumentOrShadowRoot {\n   val fullscreenElement: Element?\n   get() = definedExternally\n}\n\n/**\n * Exposes the JavaScript [ParentNode](https://developer.mozilla.org/en/docs/Web/API/ParentNode) to Kotlin\n */\npublic external interface ParentNode {\n   val children: HTMLCollection\n   val firstElementChild: Element?\n   get() = definedExternally\n   val lastElementChild: Element?\n   get() = definedExternally\n   val childElementCount: Int\n   fun prepend(vararg nodes: dynamic)\n   fun append(vararg nodes: dynamic)\n   fun querySelector(selectors: String): Element?\n   fun querySelectorAll(selectors: String): NodeList\n}\n\n/**\n * Exposes the JavaScript [NonDocumentTypeChildNode](https://developer.mozilla.org/en/docs/Web/API/NonDocumentTypeChildNode) to Kotlin\n */\npublic external interface NonDocumentTypeChildNode {\n   val previousElementSibling: Element?\n   get() = definedExternally\n   val nextElementSibling: Element?\n   get() = definedExternally\n}\n\n/**\n * Exposes the JavaScript [ChildNode](https://developer.mozilla.org/en/docs/Web/API/ChildNode) to Kotlin\n */\npublic external interface ChildNode {\n   fun before(vararg nodes: dynamic)\n   fun after(vararg nodes: dynamic)\n   fun replaceWith(vararg nodes: dynamic)\n   fun remove()\n}\n\n/**\n * Exposes the JavaScript [Slotable](https://developer.mozilla.org/en/docs/Web/API/Slotable) to Kotlin\n */\npublic external interface Slotable {\n   val assignedSlot: HTMLSlotElement?\n   get() = definedExternally\n}\n\n/**\n * Exposes the JavaScript [NodeList](https://developer.mozilla.org/en/docs/Web/API/NodeList) to Kotlin\n */\npublic external abstract class NodeList : ItemArrayLike<Node> {\n   override fun item(index: Int): Node?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun NodeList.get(index: Int): Node? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript [HTMLCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLCollection) to Kotlin\n */\npublic external abstract class HTMLCollection : ItemArrayLike<Element>, UnionElementOrHTMLCollection {\n   override fun item(index: Int): Element?\n   fun namedItem(name: String):

```
Element?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun HTMLCollection.get(index:
Int): Element? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun HTMLCollection.get(name:
String): Element? = asDynamic()[name]\n\n/**\n * Exposes the JavaScript
[MutationObserver](https://developer.mozilla.org/en/docs/Web/API/MutationObserver) to Kotlin\n */\npublic
external open class MutationObserver(callback: (Array<MutationRecord>, MutationObserver) -> Unit) {\n    fun
observe(target: Node, options: MutationObserverInit = definedExternally)\n    fun disconnect()\n    fun
takeRecords(): Array<MutationRecord>\n}\n\n/**\n * Exposes the JavaScript
[MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin\n
*/\npublic external interface MutationObserverInit {\n    var childList: Boolean? /* = false */\n        get() =
definedExternally\n        set(value) = definedExternally\n    var attributes: Boolean?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var characterData: Boolean?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var subtree: Boolean? /* = false */\n        get() =
definedExternally\n        set(value) = definedExternally\n    var attributeOldValue: Boolean?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var characterDataOldValue: Boolean?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var attributeFilter: Array<String>?\n        get() =
definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MutationObserverInit(childList:
Boolean? = false, attributes: Boolean? = undefined, characterData: Boolean? = undefined, subtree: Boolean? = false,
attributeOldValue: Boolean? = undefined, characterDataOldValue: Boolean? = undefined, attributeFilter:
Array<String>? = undefined): MutationObserverInit {\n    val o = js(\"({})\")\n    o[\"childList\"] = childList\n
o[\"attributes\"] = attributes\n    o[\"characterData\"] = characterData\n    o[\"subtree\"] = subtree\n
o[\"attributeOldValue\"] = attributeOldValue\n    o[\"characterDataOldValue\"] = characterDataOldValue\n
o[\"attributeFilter\"] = attributeFilter\n    return o\n}\n\n/**\n * Exposes the JavaScript
[MutationRecord](https://developer.mozilla.org/en/docs/Web/API/MutationRecord) to Kotlin\n */\npublic external
abstract class MutationRecord {\n    open val type: String\n    open val target: Node\n    open val addedNodes:
NodeList\n    open val removedNodes: NodeList\n    open val previousSibling: Node?\n    open val nextSibling:
Node?\n    open val attributeName: String?\n    open val attributeNamespace: String?\n    open val oldValue:
String?\n}\n\n/**\n * Exposes the JavaScript [Node](https://developer.mozilla.org/en/docs/Web/API/Node) to
Kotlin\n */\npublic external abstract class Node : EventTarget {\n    open val nodeType: Short\n    open val
nodeName: String\n    open val baseURI: String\n    open val isConnected: Boolean\n    open val ownerDocument:
Document?\n    open val parentNode: Node?\n    open val parentElement: Element?\n    open val childNodes:
NodeList\n    open val firstChild: Node?\n    open val lastChild: Node?\n    open val previousSibling: Node?\n
open val nextSibling: Node?\n    open var nodeValue: String?\n    open var textContent: String?\n    fun
getRootNode(options: GetRootNodeOptions = definedExternally): Node\n    fun hasChildNodes(): Boolean\n    fun
normalize()\n    fun cloneNode(deep: Boolean = definedExternally): Node\n    fun isEqualNode(otherNode: Node?):
Boolean\n    fun isSameNode(otherNode: Node?): Boolean\n    fun compareDocumentPosition(other: Node): Short\n
  fun contains(other: Node?): Boolean\n    fun lookupPrefix(namespace: String?): String?\n    fun
lookupNamespaceURI(prefix: String?): String?\n    fun isDefaultNamespace(namespace: String?): Boolean\n    fun
insertBefore(node: Node, child: Node?): Node\n    fun appendChild(node: Node): Node\n    fun replaceChild(node:
Node, child: Node): Node\n    fun removeChild(child: Node): Node\n\n    companion object {\n        val
ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val
CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE:
Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
```

```
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\npublic external interface
GetRootNodeOptions {\n    var composed: Boolean? /* = false */\n        get() = definedExternally\n        set(value) =
definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun GetRootNodeOptions(composed:
Boolean? = false): GetRootNodeOptions {\n    val o = js(\"({})\")\n    o[\"composed\"] = composed\n    return
o\n}\n\n/**\n * Exposes the JavaScript [Document](https://developer.mozilla.org/en/docs/Web/API/Document) to
Kotlin\n */\npublic external open class Document : Node, GlobalEventHandlers,
DocumentAndElementEventHandlers, NonElementParentNode, DocumentOrShadowRoot, ParentNode,
GeometryUtils {\n    open val implementation: DOMImplementation\n    open val URL: String\n    open val
documentURI: String\n    open val origin: String\n    open val compatMode: String\n    open val characterSet:
String\n    open val charset: String\n    open val inputEncoding: String\n    open val contentType: String\n    open val
doctype: DocumentType?\n    open val documentElement: Element?\n    open val location: Location?\n    var
domain: String\n    open val referrer: String\n    var cookie: String\n    open val lastModified: String\n    open val
readyState: DocumentReadyState\n    var title: String\n    var dir: String\n    var body: HTMLElement?\n    open val
head: HTMLHeadElement?\n    open val images: HTMLCollection\n    open val embeds: HTMLCollection\n    open
val plugins: HTMLCollection\n    open val links: HTMLCollection\n    open val forms: HTMLCollection\n    open
val scripts: HTMLCollection\n    open val currentScript: HTMLOrSVGScriptElement?\n    open val defaultView:
Window?\n    open val activeElement: Element?\n    var designMode: String\n    var onreadystatechange: ((Event) ->
dynamic)?\n    var fgColor: String\n    var linkColor: String\n    var vlinkColor: String\n    var alinkColor: String\n
var bgColor: String\n    open val anchors: HTMLCollection\n    open val applets: HTMLCollection\n    open val all:
HTMLAllCollection\n    open val scrollingElement: Element?\n    open val styleSheets: StyleSheetList\n    open val
rootElement: SVGSVGElement?\n    open val fullscreenEnabled: Boolean\n    open val fullscreen: Boolean\n    var
onfullscreenchange: ((Event) -> dynamic)?\n    var onfullscreenerror: ((Event) -> dynamic)?\n    override var
onabort: ((Event) -> dynamic)?\n    override var onblur: ((FocusEvent) -> dynamic)?\n    override var oncancel:
((Event) -> dynamic)?\n    override var oncanplay: ((Event) -> dynamic)?\n    override var oncanplaythrough:
((Event) -> dynamic)?\n    override var onchange: ((Event) -> dynamic)?\n    override var onclick: ((MouseEvent) ->
dynamic)?\n    override var onclose: ((Event) -> dynamic)?\n    override var oncontextmenu: ((MouseEvent) ->
dynamic)?\n    override var oncuechange: ((Event) -> dynamic)?\n    override var ondblclick: ((MouseEvent) ->
dynamic)?\n    override var ondrag: ((DragEvent) -> dynamic)?\n    override var ondragend: ((DragEvent) ->
dynamic)?\n    override var ondragenter: ((DragEvent) -> dynamic)?\n    override var ondragexit: ((DragEvent) ->
dynamic)?\n    override var ondragleave: ((DragEvent) -> dynamic)?\n    override var ondragover: ((DragEvent) ->
dynamic)?\n    override var ondragstart: ((DragEvent) -> dynamic)?\n    override var ondrop: ((DragEvent) ->
dynamic)?\n    override var ondurationchange: ((Event) -> dynamic)?\n    override var onemptied: ((Event) ->
dynamic)?\n    override var onended: ((Event) -> dynamic)?\n    override var onerror: ((dynamic, String, Int, Int,
Any?) -> dynamic)?\n    override var onfocus: ((FocusEvent) -> dynamic)?\n    override var oninput: ((InputEvent) -
> dynamic)?\n    override var oninvalid: ((Event) -> dynamic)?\n    override var onkeydown: ((KeyboardEvent) ->
dynamic)?\n    override var onkeypress: ((KeyboardEvent) -> dynamic)?\n    override var onkeyup:
((KeyboardEvent) -> dynamic)?\n    override var onload: ((Event) -> dynamic)?\n    override var onloadeddata:
((Event) -> dynamic)?\n    override var onloadedmetadata: ((Event) -> dynamic)?\n    override var onloadend:
((Event) -> dynamic)?\n    override var onloadstart: ((ProgressEvent) -> dynamic)?\n    override var onmousedown:
((MouseEvent) -> dynamic)?\n    override var onmouseenter: ((MouseEvent) -> dynamic)?\n    override var
onmouseleave: ((MouseEvent) -> dynamic)?\n    override var onmousemove: ((MouseEvent) -> dynamic)?\n
override var onmouseout: ((MouseEvent) -> dynamic)?\n    override var onmouseover: ((MouseEvent) ->
dynamic)?\n    override var onmouseup: ((MouseEvent) -> dynamic)?\n    override var onwheel: ((WheelEvent) ->
dynamic)?\n    override var onpause: ((Event) -> dynamic)?\n    override var onplay: ((Event) -> dynamic)?\n
override var onplaying: ((Event) -> dynamic)?\n    override var onprogress: ((ProgressEvent) -> dynamic)?\n
```

override var onratechange: ((Event) -> dynamic)?\n   override var onreset: ((Event) -> dynamic)?\n   override var onresize: ((Event) -> dynamic)?\n   override var onscroll: ((Event) -> dynamic)?\n   override var onseeked: ((Event) -> dynamic)?\n   override var onseeking: ((Event) -> dynamic)?\n   override var onselect: ((Event) -> dynamic)?\n   override var onshow: ((Event) -> dynamic)?\n   override var onstalled: ((Event) -> dynamic)?\n   override var onsubmit: ((Event) -> dynamic)?\n   override var onsuspend: ((Event) -> dynamic)?\n   override var ontimeupdate: ((Event) -> dynamic)?\n   override var ontoggle: ((Event) -> dynamic)?\n   override var onvolumechange: ((Event) -> dynamic)?\n   override var onwaiting: ((Event) -> dynamic)?\n   override var ongotpointercapture: ((PointerEvent) -> dynamic)?\n   override var onlostpointercapture: ((PointerEvent) -> dynamic)?\n   override var onpointerdown: ((PointerEvent) -> dynamic)?\n   override var onpointermove: ((PointerEvent) -> dynamic)?\n   override var onpointerup: ((PointerEvent) -> dynamic)?\n   override var onpointercancel: ((PointerEvent) -> dynamic)?\n   override var onpointerover: ((PointerEvent) -> dynamic)?\n   override var onpointerout: ((PointerEvent) -> dynamic)?\n   override var onpointerenter: ((PointerEvent) -> dynamic)?\n   override var onpointerleave: ((PointerEvent) -> dynamic)?\n   override var oncopy: ((ClipboardEvent) -> dynamic)?\n   override var oncut: ((ClipboardEvent) -> dynamic)?\n   override var onpaste: ((ClipboardEvent) -> dynamic)?\n   override val fullscreenElement: Element?\n   override val children: HTMLCollection\n   override val firstElementChild: Element?\n   override val lastElementChild: Element?\n   override val childElementCount: Int\n   fun getElementsByTagName(qualifiedName: String): HTMLCollection\n   fun getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection\n   fun getElementsByClassName(classNames: String): HTMLCollection\n   fun createElement(localName: String, options: ElementCreationOptions = definedExternally): Element\n   fun createElementNS(namespace: String?, qualifiedName: String, options: ElementCreationOptions = definedExternally): Element\n   fun createDocumentFragment(): DocumentFragment\n   fun createTextNode(data: String): Text\n   fun createCDATASection(data: String): CDATASection\n   fun createComment(data: String): Comment\n   fun createProcessingInstruction(target: String, data: String): ProcessingInstruction\n   fun importNode(node: Node, deep: Boolean = definedExternally): Node\n   fun adoptNode(node: Node): Node\n   fun createAttribute(localName: String): Attr\n   fun createAttributeNS(namespace: String?, qualifiedName: String): Attr\n   fun createEvent(`interface`: String): Event\n   fun createRange(): Range\n   fun createNodeIterator(root: Node, whatToShow: Int = definedExternally, filter: NodeFilter? = definedExternally): NodeIterator\n   fun createNodeIterator(root: Node, whatToShow: Int = definedExternally, filter: ((Node) -> Short)? = definedExternally): NodeIterator\n   fun createTreeWalker(root: Node, whatToShow: Int = definedExternally, filter: NodeFilter? = definedExternally): TreeWalker\n   fun createTreeWalker(root: Node, whatToShow: Int = definedExternally, filter: ((Node) -> Short)? = definedExternally): TreeWalker\n   fun getElementsByName(elementName: String): NodeList\n   fun open(type: String = definedExternally, replace: String = definedExternally): Document\n   fun open(url: String, name: String, features: String): Window\n   fun close()\n   fun write(vararg text: String)\n   fun writeln(vararg text: String)\n   fun hasFocus(): Boolean\n   fun execCommand(commandId: String, showUI: Boolean = definedExternally, value: String = definedExternally): Boolean\n   fun queryCommandEnabled(commandId: String): Boolean\n   fun queryCommandIndeterm(commandId: String): Boolean\n   fun queryCommandState(commandId: String): Boolean\n   fun queryCommandSupported(commandId: String): Boolean\n   fun queryCommandValue(commandId: String): String\n   fun clear()\n   fun captureEvents()\n   fun releaseEvents()\n   fun elementFromPoint(x: Double, y: Double): Element?\n   fun elementsFromPoint(x: Double, y: Double): Array<Element>\n   fun caretPositionFromPoint(x: Double, y: Double): CaretPosition?\n   fun createTouch(view: Window, target: EventTarget, identifier: Int, pageX: Int, pageY: Int, screenX: Int, screenY: Int): Touch\n   fun createTouchList(vararg touches: Touch): TouchList\n   fun exitFullscreen(): Promise<Unit>\n   override fun getElementById(elementId: String): Element?\n   override fun prepend(vararg nodes: dynamic)\n   override fun append(vararg nodes: dynamic)\n   override fun querySelector(selectors: String): Element?\n   override fun querySelectorAll(selectors: String): NodeList\n   override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>\n   override fun convertQuadFromNode(quad: dynamic, from: dynamic,

```
options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n    override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n    override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint\n\n    companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n        val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n        val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun Document.get(name: String): dynamic = asDynamic()[name]\n\n/**\n * Exposes the JavaScript [XMLDocument](https://developer.mozilla.org/en/docs/Web/API/XMLDocument) to Kotlin\n */\npublic external open class XMLDocument : Document {\n    companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n        val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n        val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\npublic external interface ElementCreationOptions {\n    var `is`: String?\n        get() = definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ElementCreationOptions(`is`: String? = undefined): ElementCreationOptions {\n    val o = js(\"({})\")\n    o[\"is\"] = `is`\n    return o\n}\n\n/**\n * Exposes the JavaScript [DOMImplementation](https://developer.mozilla.org/en/docs/Web/API/DOMImplementation) to Kotlin\n */\npublic external abstract class DOMImplementation {\n    fun createDocumentType(qualifiedName: String, publicId: String, systemId: String): DocumentType\n    fun createDocument(namespace: String?, qualifiedName: String, doctype: DocumentType? = definedExternally): XMLDocument\n    fun createHTMLDocument(title: String = definedExternally): Document\n    fun hasFeature(): Boolean\n}\n\n/**\n * Exposes the JavaScript [DocumentType](https://developer.mozilla.org/en/docs/Web/API/DocumentType) to Kotlin\n */\npublic external abstract class DocumentType : Node, ChildNode {\n    open val name: String\n    open val publicId: String\n    open val systemId: String\n\n    companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n        val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n        val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
```

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [DocumentFragment](https://developer.mozilla.org/en/docs/Web/API/DocumentFragment) to Kotlin\n */\npublic external open class DocumentFragment : Node, NonElementParentNode, ParentNode {\n override val children: HTMLCollection\n override val firstElementChild: Element?\n override val lastElementChild: Element?\n override val childElementCount: Int\n override fun getElementById(elementId: String): Element?\n override fun prepend(vararg nodes: dynamic)\n override fun append(vararg nodes: dynamic)\n override fun querySelector(selectors: String): Element?\n override fun querySelectorAll(selectors: String): NodeList\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [ShadowRoot](https://developer.mozilla.org/en/docs/Web/API/ShadowRoot) to Kotlin\n */\npublic external open class ShadowRoot : DocumentFragment, DocumentOrShadowRoot {\n open val mode: ShadowRootMode\n open val host: Element\n override val fullscreenElement: Element?\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [Element](https://developer.mozilla.org/en/docs/Web/API/Element) to Kotlin\n */\npublic external abstract class Element : Node, ParentNode, NonDocumentTypeChildNode, ChildNode, Slotable, GeometryUtils, UnionElementOrHTMLCollection, UnionElementOrRadioNodeList, UnionElementOrMouseEvent, UnionElementOrProcessingInstruction {\n open val namespaceURI: String?\n open val prefix: String?\n open val localName: String\n open val tagName: String\n open var id: String\n open var className: String\n open val classList: DOMTokenList\n open var slot: String\n open val attributes: NamedNodeMap\n open val shadowRoot: ShadowRoot?\n open var scrollTop: Double\n open var scrollLeft: Double\n open val scrollWidth: Int\n open val scrollHeight: Int\n open val clientTop: Int\n open val clientLeft: Int\n open val clientWidth: Int\n open val clientHeight: Int\n open var innerHTML: String\n open var outerHTML: String\n fun hasAttributes(): Boolean\n fun getAttributeNames(): Array<String>\n fun getAttribute(qualifiedName: String): String?\n fun getAttributeNS(namespace: String?, localName: String): String?\n fun setAttribute(qualifiedName: String, value: String)\n fun setAttributeNS(namespace: String?, qualifiedName: String, value: String)\n fun removeAttribute(qualifiedName: String)\n fun removeAttributeNS(namespace: String?, localName: String)\n fun hasAttribute(qualifiedName: String): Boolean\n fun hasAttributeNS(namespace: String?, localName: String): Boolean\n fun getAttributeNode(qualifiedName: String): Attr?\n fun getAttributeNodeNS(namespace: String?, localName: String): Attr?\n fun setAttributeNode(attr: Attr): Attr?\n fun setAttributeNodeNS(attr: Attr): Attr?\n fun removeAttributeNode(attr: Attr): Attr\n fun attachShadow(init: ShadowRootInit): ShadowRoot\n fun closest(selectors: String): Element?\n fun matches(selectors: String): Boolean\n fun webkitMatchesSelector(selectors: String): Boolean\n fun getElementsByTagName(qualifiedName: String): HTMLCollection\n fun

```
getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection\n    fun
getElementsByClassName(classNames: String): HTMLCollection\n    fun insertAdjacentElement(where: String,
element: Element): Element?\n    fun insertAdjacentText(where: String, data: String)\n    fun getClientRects():
Array<DOMRect>\n    fun getBoundingClientRect(): DOMRect\n    fun scrollIntoView()\n    fun
scrollIntoView(arg: dynamic)\n    fun scroll(options: ScrollToOptions = definedExternally)\n    fun scroll(x: Double,
y: Double)\n    fun scrollTo(options: ScrollToOptions = definedExternally)\n    fun scrollTo(x: Double, y: Double)\n
  fun scrollBy(options: ScrollToOptions = definedExternally)\n    fun scrollBy(x: Double, y: Double)\n    fun
insertAdjacentHTML(position: String, text: String)\n    fun setPointerCapture(pointerId: Int)\n    fun
releasePointerCapture(pointerId: Int)\n    fun hasPointerCapture(pointerId: Int): Boolean\n    fun requestFullscreen():
Promise<Unit>\n\n    companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE:
Short\n        val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val
ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE: Short\n        val
PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
     val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\npublic external interface
ShadowRootInit {\n    var mode: ShadowRootMode?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ShadowRootInit(mode:
ShadowRootMode?): ShadowRootInit {\n    val o = js(\"({})\")\n    o[\"mode\"] = mode\n    return o\n}\n\n/**\n *
Exposes the JavaScript [NamedNodeMap](https://developer.mozilla.org/en/docs/Web/API/NamedNodeMap) to
Kotlin\n */\npublic external abstract class NamedNodeMap : ItemArrayLike<Attr> {\n    fun
getNamedItemNS(namespace: String?, localName: String): Attr?\n    fun setNamedItem(attr: Attr): Attr?\n    fun
setNamedItemNS(attr: Attr): Attr?\n    fun removeNamedItem(qualifiedName: String): Attr\n    fun
removeNamedItemNS(namespace: String?, localName: String): Attr\n    override fun item(index: Int): Attr?\n    fun
getNamedItem(qualifiedName: String): Attr?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun NamedNodeMap.get(index:
Int): Attr? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun
NamedNodeMap.get(qualifiedName: String): Attr? = asDynamic()[qualifiedName]\n\n/**\n * Exposes the
JavaScript [Attr](https://developer.mozilla.org/en/docs/Web/API/Attr) to Kotlin\n */\npublic external abstract class
Attr : Node {\n    open val namespaceURI: String?\n    open val prefix: String?\n    open val localName: String\n
open val name: String\n    open var value: String\n    open val ownerElement: Element?\n    open val specified:
Boolean\n\n    companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n
val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE:
Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val
COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n
     val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
     val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[CharacterData](https://developer.mozilla.org/en/docs/Web/API/CharacterData) to Kotlin\n */\npublic external
abstract class CharacterData : Node, NonDocumentTypeChildNode, ChildNode {\n    open var data: String\n    open
val length: Int\n    fun substringData(offset: Int, count: Int): String\n    fun appendData(data: String)\n    fun
insertData(offset: Int, data: String)\n    fun deleteData(offset: Int, count: Int)\n    fun replaceData(offset: Int, count:
```

Int, data: String)\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [Text](https://developer.mozilla.org/en/docs/Web/API/Text) to Kotlin\n */\npublic external open class Text(data: String = definedExternally) : CharacterData, Slotable, GeometryUtils {\n   open val wholeText: String\n   override val assignedSlot: HTMLSlotElement?\n   override val previousElementSibling: Element?\n   override val nextElementSibling: Element?\n   fun splitText(offset: Int): Text\n   override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>\n   override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n   override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n   override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint\n   override fun before(vararg nodes: dynamic)\n   override fun after(vararg nodes: dynamic)\n   override fun replaceWith(vararg nodes: dynamic)\n   override fun remove()\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [CDATASection](https://developer.mozilla.org/en/docs/Web/API/CDATASection) to Kotlin\n */\npublic external open class CDATASection : Text {\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [ProcessingInstruction](https://developer.mozilla.org/en/docs/Web/API/ProcessingInstruction) to Kotlin\n */\npublic external abstract class ProcessingInstruction : CharacterData, LinkStyle, UnionElementOrProcessingInstruction {\n   open val target: String\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val

```
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[Comment](https://developer.mozilla.org/en/docs/Web/API/Comment) to Kotlin\n */\npublic external open class
Comment(data: String = definedExternally) : CharacterData {\n    override val previousElementSibling: Element?\n
 override val nextElementSibling: Element?\n    override fun before(vararg nodes: dynamic)\n    override fun
after(vararg nodes: dynamic)\n    override fun replaceWith(vararg nodes: dynamic)\n    override fun remove()\n\n
companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val
TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE:
Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val
COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n
    val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[Range](https://developer.mozilla.org/en/docs/Web/API/Range) to Kotlin\n */\npublic external open class Range {\n
  open val startContainer: Node\n    open val startOffset: Int\n    open val endContainer: Node\n    open val
endOffset: Int\n    open val collapsed: Boolean\n    open val commonAncestorContainer: Node\n    fun setStart(node:
Node, offset: Int)\n    fun setEnd(node: Node, offset: Int)\n    fun setStartBefore(node: Node)\n    fun
setStartAfter(node: Node)\n    fun setEndBefore(node: Node)\n    fun setEndAfter(node: Node)\n    fun
collapse(toStart: Boolean = definedExternally)\n    fun selectNode(node: Node)\n    fun selectNodeContents(node:
Node)\n    fun compareBoundaryPoints(how: Short, sourceRange: Range): Short\n    fun deleteContents()\n    fun
extractContents(): DocumentFragment\n    fun cloneContents(): DocumentFragment\n    fun insertNode(node:
Node)\n    fun surroundContents(newParent: Node)\n    fun cloneRange(): Range\n    fun detach()\n    fun
isPointInRange(node: Node, offset: Int): Boolean\n    fun comparePoint(node: Node, offset: Int): Short\n    fun
intersectsNode(node: Node): Boolean\n    fun getClientRects(): Array<DOMRect>\n    fun
getBoundingClientRect(): DOMRect\n    fun createContextualFragment(fragment: String): DocumentFragment\n\n
companion object {\n        val START_TO_START: Short\n        val START_TO_END: Short\n        val
END_TO_END: Short\n        val END_TO_START: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[NodeIterator](https://developer.mozilla.org/en/docs/Web/API/NodeIterator) to Kotlin\n */\npublic external abstract
class NodeIterator {\n    open val root: Node\n    open val referenceNode: Node\n    open val
pointerBeforeReferenceNode: Boolean\n    open val whatToShow: Int\n    open val filter: NodeFilter?\n    fun
nextNode(): Node?\n    fun previousNode(): Node?\n    fun detach()\n}\n\n/**\n * Exposes the JavaScript
[TreeWalker](https://developer.mozilla.org/en/docs/Web/API/TreeWalker) to Kotlin\n */\npublic external abstract
class TreeWalker {\n    open val root: Node\n    open val whatToShow: Int\n    open val filter: NodeFilter?\n    open
var currentNode: Node\n    fun parentNode(): Node?\n    fun firstChild(): Node?\n    fun lastChild(): Node?\n    fun
previousSibling(): Node?\n    fun nextSibling(): Node?\n    fun previousNode(): Node?\n    fun nextNode():
Node?\n}\n\n/**\n * Exposes the JavaScript
[NodeFilter](https://developer.mozilla.org/en/docs/Web/API/NodeFilter) to Kotlin\n
*/\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface NodeFilter {\n
fun acceptNode(node: Node): Short\n\n    companion object {\n        val FILTER_ACCEPT: Short\n        val
FILTER_REJECT: Short\n        val FILTER_SKIP: Short\n        val SHOW_ALL: Int\n        val
SHOW_ELEMENT: Int\n        val SHOW_ATTRIBUTE: Int\n        val SHOW_TEXT: Int\n        val
SHOW_CDATA_SECTION: Int\n        val SHOW_ENTITY_REFERENCE: Int\n        val SHOW_ENTITY: Int\n
    val SHOW_PROCESSING_INSTRUCTION: Int\n        val SHOW_COMMENT: Int\n        val
```

SHOW_DOCUMENT: Int\n val SHOW_DOCUMENT_TYPE: Int\n val SHOW_DOCUMENT_FRAGMENT: Int\n val SHOW_NOTATION: Int\n }\n}\n\n/**\n * Exposes the JavaScript [DOMTokenList](https://developer.mozilla.org/en/docs/Web/API/DOMTokenList) to Kotlin\n */\npublic external abstract class DOMTokenList : ItemArrayLike<String> {\n open var value: String\n fun contains(token: String): Boolean\n fun add(vararg tokens: String)\n fun remove(vararg tokens: String)\n fun toggle(token: String, force: Boolean = definedExternally): Boolean\n fun replace(token: String, newToken: String)\n fun supports(token: String): Boolean\n override fun item(index: Int): String?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun DOMTokenList.get(index: Int): String? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript [DOMPointReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMPointReadOnly) to Kotlin\n */\npublic external open class DOMPointReadOnly(x: Double, y: Double, z: Double, w: Double) {\n open val x: Double\n open val y: Double\n open val z: Double\n open val w: Double\n fun matrixTransform(matrix: DOMMatrixReadOnly): DOMPoint\n}\n\n/**\n * Exposes the JavaScript [DOMPoint](https://developer.mozilla.org/en/docs/Web/API/DOMPoint) to Kotlin\n */\npublic external open class DOMPoint : DOMPointReadOnly {\n constructor(point: DOMPointInit)\n constructor(x: Double = definedExternally, y: Double = definedExternally, z: Double = definedExternally, w: Double = definedExternally)\n override var x: Double\n override var y: Double\n override var z: Double\n override var w: Double\n}\n\n/**\n * Exposes the JavaScript [DOMPointInit](https://developer.mozilla.org/en/docs/Web/API/DOMPointInit) to Kotlin\n */\npublic external interface DOMPointInit {\n var x: Double? /* = 0.0 */\n get() = definedExternally\n set(value) = definedExternally\n var y: Double? /* = 0.0 */\n get() = definedExternally\n set(value) = definedExternally\n var z: Double? /* = 0.0 */\n get() = definedExternally\n set(value) = definedExternally\n var w: Double? /* = 1.0 */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun DOMPointInit(x: Double? = 0.0, y: Double? = 0.0, z: Double? = 0.0, w: Double? = 1.0): DOMPointInit {\n val o = js(\"({})\")\n o[\"x\"] = x\n o[\"y\"] = y\n o[\"z\"] = z\n o[\"w\"] = w\n return o\n}\n\n/**\n * Exposes the JavaScript [DOMRect](https://developer.mozilla.org/en/docs/Web/API/DOMRect) to Kotlin\n */\npublic external open class DOMRect(x: Double = definedExternally, y: Double = definedExternally, width: Double = definedExternally, height: Double = definedExternally) : DOMRectReadOnly {\n override var x: Double\n override var y: Double\n override var width: Double\n override var height: Double\n}\n\n/**\n * Exposes the JavaScript [DOMRectReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMRectReadOnly) to Kotlin\n */\npublic external open class DOMRectReadOnly(x: Double, y: Double, width: Double, height: Double) {\n open val x: Double\n open val y: Double\n open val width: Double\n open val height: Double\n open val top: Double\n open val right: Double\n open val bottom: Double\n open val left: Double\n}\n\npublic external interface DOMRectInit {\n var x: Double? /* = 0.0 */\n get() = definedExternally\n set(value) = definedExternally\n var y: Double? /* = 0.0 */\n get() = definedExternally\n set(value) = definedExternally\n var width: Double? /* = 0.0 */\n get() = definedExternally\n set(value) = definedExternally\n var height: Double? /* = 0.0 */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun DOMRectInit(x: Double? = 0.0, y: Double? = 0.0, width: Double? = 0.0, height: Double? = 0.0): DOMRectInit {\n val o = js(\"({})\")\n o[\"x\"] = x\n o[\"y\"] = y\n o[\"width\"] = width\n o[\"height\"] = height\n return o\n}\n\npublic external interface DOMRectList : ItemArrayLike<DOMRect> {\n override fun item(index: Int): DOMRect?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun DOMRectList.get(index: Int): DOMRect? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript

[DOMQuad](https://developer.mozilla.org/en/docs/Web/API/DOMQuad) to Kotlin\n */\npublic external open class DOMQuad {\n constructor(p1: DOMPointInit = definedExternally, p2: DOMPointInit = definedExternally, p3: DOMPointInit = definedExternally, p4: DOMPointInit = definedExternally)\n constructor(rect: DOMRectInit)\n open val p1: DOMPoint\n open val p2: DOMPoint\n open val p3: DOMPoint\n open val p4: DOMPoint\n open val bounds: DOMRectReadOnly\n}\n\n/**\n * Exposes the JavaScript [DOMMatrixReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMMatrixReadOnly) to Kotlin\n */\npublic external open class DOMMatrixReadOnly(numberSequence: Array<Double>) {\n open val a: Double\n open val b: Double\n open val c: Double\n open val d: Double\n open val e: Double\n open val f: Double\n open val m11: Double\n open val m12: Double\n open val m13: Double\n open val m14: Double\n open val m21: Double\n open val m22: Double\n open val m23: Double\n open val m24: Double\n open val m31: Double\n open val m32: Double\n open val m33: Double\n open val m34: Double\n open val m41: Double\n open val m42: Double\n open val m43: Double\n open val m44: Double\n open val is2D: Boolean\n open val isIdentity: Boolean\n fun translate(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix\n fun scale(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix\n fun scale3d(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix\n fun scaleNonUniform(scaleX: Double, scaleY: Double = definedExternally, scaleZ: Double = definedExternally, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix\n fun rotate(angle: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix\n fun rotateFromVector(x: Double, y: Double): DOMMatrix\n fun rotateAxisAngle(x: Double, y: Double, z: Double, angle: Double): DOMMatrix\n fun skewX(sx: Double): DOMMatrix\n fun skewY(sy: Double): DOMMatrix\n fun multiply(other: DOMMatrix): DOMMatrix\n fun flipX(): DOMMatrix\n fun flipY(): DOMMatrix\n fun inverse(): DOMMatrix\n fun transformPoint(point: DOMPointInit = definedExternally): DOMPoint\n fun toFloat32Array(): Float32Array\n fun toFloat64Array(): Float64Array\n}\n\n/**\n * Exposes the JavaScript [DOMMatrix](https://developer.mozilla.org/en/docs/Web/API/DOMMatrix) to Kotlin\n */\npublic external open class DOMMatrix() : DOMMatrixReadOnly {\n constructor(transformList: String)\n constructor(other: DOMMatrixReadOnly)\n constructor(array32: Float32Array)\n constructor(array64: Float64Array)\n constructor(numberSequence: Array<Double>)\n override var a: Double\n override var b: Double\n override var c: Double\n override var d: Double\n override var e: Double\n override var f: Double\n override var m11: Double\n override var m12: Double\n override var m13: Double\n override var m14: Double\n override var m21: Double\n override var m22: Double\n override var m23: Double\n override var m24: Double\n override var m31: Double\n override var m32: Double\n override var m33: Double\n override var m34: Double\n override var m41: Double\n override var m42: Double\n override var m43: Double\n override var m44: Double\n fun multiplySelf(other: DOMMatrix): DOMMatrix\n fun preMultiplySelf(other: DOMMatrix): DOMMatrix\n fun translateSelf(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix\n fun scaleSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix\n fun scale3dSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix\n fun scaleNonUniformSelf(scaleX: Double, scaleY: Double = definedExternally, scaleZ: Double = definedExternally, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix\n fun rotateSelf(angle: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix\n fun rotateFromVectorSelf(x: Double, y: Double): DOMMatrix\n fun rotateAxisAngleSelf(x: Double, y: Double, z: Double, angle: Double): DOMMatrix\n fun skewXSelf(sx: Double): DOMMatrix\n fun skewYSelf(sy: Double): DOMMatrix\n fun invertSelf(): DOMMatrix\n fun setMatrixValue(transformList: String): DOMMatrix\n}\n\npublic external interface ScrollOptions {\n var behavior: ScrollBehavior? /* = ScrollBehavior.AUTO */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ScrollOptions(behavior: ScrollBehavior? = ScrollBehavior.AUTO): ScrollOptions {\n val o = js(\"({})\")\n o[\"behavior\"] = behavior\n

return o\n}\n\n/**\n * Exposes the JavaScript [ScrollToOptions](https://developer.mozilla.org/en/docs/Web/API/ScrollToOptions) to Kotlin\n */\npublic external interface ScrollToOptions : ScrollOptions {\n var left: Double?\n get() = definedExternally\n set(value) = definedExternally\n var top: Double?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ScrollToOptions(left: Double? = undefined, top: Double? = undefined, behavior: ScrollBehavior? = ScrollBehavior.AUTO): ScrollToOptions {\n val o = js(\"({})\")\n o[\"left\"] = left\n o[\"top\"] = top\n o[\"behavior\"] = behavior\n return o\n}\n\n/**\n * Exposes the JavaScript [MediaQueryList](https://developer.mozilla.org/en/docs/Web/API/MediaQueryList) to Kotlin\n */\npublic external abstract class MediaQueryList : EventTarget {\n open val media: String\n open val matches: Boolean\n open var onchange: ((Event) -> dynamic)?\n fun addListener(listener: EventListener?)\n fun addListener(listener: ((Event) -> Unit)?)\n fun removeListener(listener: EventListener?)\n fun removeListener(listener: ((Event) -> Unit)?)\n}\n\n/**\n * Exposes the JavaScript [MediaQueryListEvent](https://developer.mozilla.org/en/docs/Web/API/MediaQueryListEvent) to Kotlin\n */\npublic external open class MediaQueryListEvent(type: String, eventInitDict: MediaQueryListEventInit = definedExternally) : Event {\n open val media: String\n open val matches: Boolean\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface MediaQueryListEventInit : EventInit {\n var media: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n var matches: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaQueryListEventInit(media: String? = \"\", matches: Boolean? = false, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): MediaQueryListEventInit {\n val o = js(\"({})\")\n o[\"media\"] = media\n o[\"matches\"] = matches\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [Screen](https://developer.mozilla.org/en/docs/Web/API/Screen) to Kotlin\n */\npublic external abstract class Screen {\n open val availWidth: Int\n open val availHeight: Int\n open val width: Int\n open val height: Int\n open val colorDepth: Int\n open val pixelDepth: Int\n}\n\n/**\n * Exposes the JavaScript [CaretPosition](https://developer.mozilla.org/en/docs/Web/API/CaretPosition) to Kotlin\n */\npublic external abstract class CaretPosition {\n open val offsetNode: Node\n open val offset: Int\n fun getClientRect(): DOMRect?\n}\n\npublic external interface ScrollIntoViewOptions : ScrollOptions {\n var block: ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */\n get() = definedExternally\n set(value) = definedExternally\n var inline: ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ScrollIntoViewOptions(block: ScrollLogicalPosition? = ScrollLogicalPosition.CENTER, inline: ScrollLogicalPosition? = ScrollLogicalPosition.CENTER, behavior: ScrollBehavior? = ScrollBehavior.AUTO): ScrollIntoViewOptions {\n val o = js(\"({})\")\n o[\"block\"] = block\n o[\"inline\"] = inline\n o[\"behavior\"] = behavior\n return o\n}\n\npublic external interface BoxQuadOptions {\n var box: CSSBoxType? /* = CSSBoxType.BORDER */\n get() = definedExternally\n set(value) = definedExternally\n var relativeTo: dynamic\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun BoxQuadOptions(box: CSSBoxType? = CSSBoxType.BORDER, relativeTo: dynamic = undefined): BoxQuadOptions {\n val o = js(\"({})\")\n o[\"box\"] = box\n o[\"relativeTo\"] = relativeTo\n return o\n}\n\npublic external interface ConvertCoordinateOptions {\n var fromBox: CSSBoxType? /* = CSSBoxType.BORDER */\n get() = definedExternally\n set(value) = definedExternally\n var toBox: CSSBoxType? /* = CSSBoxType.BORDER */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",

\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ConvertCoordinateOptions(fromBox: CSSBoxType? = CSSBoxType.BORDER, toBox: CSSBoxType? = CSSBoxType.BORDER): ConvertCoordinateOptions {\n   val o = js(\"({})\")\n   o[\"fromBox\"] = fromBox\n   o[\"toBox\"] = toBox\n   return o\n}\n\n/**\n * Exposes the JavaScript [GeometryUtils](https://developer.mozilla.org/en/docs/Web/API/GeometryUtils) to Kotlin\n */\npublic external interface GeometryUtils {\n   fun getBoxQuads(options: BoxQuadOptions = definedExternally): Array<DOMQuad>\n   fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMQuad\n   fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMQuad\n   fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMPoint\n}\n\n/**\n * Exposes the JavaScript [Touch](https://developer.mozilla.org/en/docs/Web/API/Touch) to Kotlin\n */\npublic external abstract class Touch {\n   open val identifier: Int\n   open val target: EventTarget\n   open val screenX: Int\n   open val screenY: Int\n   open val clientX: Int\n   open val clientY: Int\n   open val pageX: Int\n   open val pageY: Int\n   open val region: String?\n}\n\npublic external abstract class TouchList : ItemArrayLike<Touch> {\n   override fun item(index: Int): Touch?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun TouchList.get(index: Int): Touch? = asDynamic()[index]\n\npublic external open class TouchEvent : UIEvent {\n   open val touches: TouchList\n   open val targetTouches: TouchList\n   open val changedTouches: TouchList\n   open val altKey: Boolean\n   open val metaKey: Boolean\n   open val ctrlKey: Boolean\n   open val shiftKey: Boolean\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [Image](https://developer.mozilla.org/en/docs/Web/API/Image) to Kotlin\n */\npublic external open class Image(width: Int = definedExternally, height: Int = definedExternally) : HTMLImageElement {\n   override var onabort: ((Event) -> dynamic)?\n   override var onblur: ((FocusEvent) -> dynamic)?\n   override var oncancel: ((Event) -> dynamic)?\n   override var oncanplay: ((Event) -> dynamic)?\n   override var oncanplaythrough: ((Event) -> dynamic)?\n   override var onchange: ((Event) -> dynamic)?\n   override var onclick: ((MouseEvent) -> dynamic)?\n   override var onclose: ((Event) -> dynamic)?\n   override var oncontextmenu: ((MouseEvent) -> dynamic)?\n   override var oncuechange: ((Event) -> dynamic)?\n   override var ondblclick: ((MouseEvent) -> dynamic)?\n   override var ondrag: ((DragEvent) -> dynamic)?\n   override var ondragend: ((DragEvent) -> dynamic)?\n   override var ondragenter: ((DragEvent) -> dynamic)?\n   override var ondragexit: ((DragEvent) -> dynamic)?\n   override var ondragleave: ((DragEvent) -> dynamic)?\n   override var ondragover: ((DragEvent) -> dynamic)?\n   override var ondragstart: ((DragEvent) -> dynamic)?\n   override var ondrop: ((DragEvent) -> dynamic)?\n   override var ondurationchange: ((Event) -> dynamic)?\n   override var onemptied: ((Event) -> dynamic)?\n   override var onended: ((Event) -> dynamic)?\n   override var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?\n   override var onfocus: ((FocusEvent) -> dynamic)?\n   override var oninput: ((InputEvent) -> dynamic)?\n   override var oninvalid: ((Event) -> dynamic)?\n   override var onkeydown: ((KeyboardEvent) -> dynamic)?\n   override var onkeypress: ((KeyboardEvent) -> dynamic)?\n   override var onkeyup: ((KeyboardEvent) -> dynamic)?\n   override var onload: ((Event) -> dynamic)?\n   override var onloadeddata: ((Event) -> dynamic)?\n   override var onloadedmetadata: ((Event) -> dynamic)?\n   override var onloadend: ((Event) -> dynamic)?\n   override var onloadstart: ((ProgressEvent) -> dynamic)?\n   override var onmousedown: ((MouseEvent) -> dynamic)?\n   override var onmouseenter: ((MouseEvent) -> dynamic)?\n   override var onmouseleave: ((MouseEvent) -> dynamic)?\n   override var onmousemove: ((MouseEvent) -> dynamic)?\n   override var onmouseout: ((MouseEvent) -> dynamic)?\n   override var onmouseover: ((MouseEvent) -> dynamic)?\n   override var onmouseup: ((MouseEvent) -> dynamic)?\n   override var onwheel: ((WheelEvent) -> dynamic)?\n   override var onpause: ((Event) -> dynamic)?\n   override var onplay: ((Event) -> dynamic)?\n   override var onplaying: ((Event) -> dynamic)?\n   override var onprogress: ((ProgressEvent) -> dynamic)?\n   override var onratechange: ((Event) -> dynamic)?\n   override var onreset: ((Event) -> dynamic)?\n   override var

onresize: ((Event) -> dynamic)?\n override var onscroll: ((Event) -> dynamic)?\n override var onseeked: ((Event) -> dynamic)?\n override var onseeking: ((Event) -> dynamic)?\n override var onselect: ((Event) -> dynamic)?\n override var onshow: ((Event) -> dynamic)?\n override var onstalled: ((Event) -> dynamic)?\n override var onsubmit: ((Event) -> dynamic)?\n override var onsuspend: ((Event) -> dynamic)?\n override var ontimeupdate: ((Event) -> dynamic)?\n override var ontoggle: ((Event) -> dynamic)?\n override var onvolumechange: ((Event) -> dynamic)?\n override var onwaiting: ((Event) -> dynamic)?\n override var ongotpointercapture: ((PointerEvent) -> dynamic)?\n override var onlostpointercapture: ((PointerEvent) -> dynamic)?\n override var onpointerdown: ((PointerEvent) -> dynamic)?\n override var onpointermove: ((PointerEvent) -> dynamic)?\n override var onpointerup: ((PointerEvent) -> dynamic)?\n override var onpointercancel: ((PointerEvent) -> dynamic)?\n override var onpointerover: ((PointerEvent) -> dynamic)?\n override var onpointerout: ((PointerEvent) -> dynamic)?\n override var onpointerenter: ((PointerEvent) -> dynamic)?\n override var onpointerleave: ((PointerEvent) -> dynamic)?\n override var oncopy: ((ClipboardEvent) -> dynamic)?\n override var oncut: ((ClipboardEvent) -> dynamic)?\n override var onpaste: ((ClipboardEvent) -> dynamic)?\n override var contentEditable: String\n override val isContentEditable: Boolean\n override val style: CSSStyleDeclaration\n override val children: HTMLCollection\n override val firstElementChild: Element?\n override val lastElementChild: Element?\n override val childElementCount: Int\n override val previousElementSibling: Element?\n override val nextElementSibling: Element?\n override val assignedSlot: HTMLSlotElement?\n override fun prepend(vararg nodes: dynamic)\n override fun append(vararg nodes: dynamic)\n override fun querySelector(selectors: String): Element?\n override fun querySelectorAll(selectors: String): NodeList\n override fun before(vararg nodes: dynamic)\n override fun after(vararg nodes: dynamic)\n override fun replaceWith(vararg nodes: dynamic)\n override fun remove()\n override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>\n override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external open class Audio(src: String = definedExternally) : HTMLAudioElement {\n override var onabort: ((Event) -> dynamic)?\n override var onblur: ((FocusEvent) -> dynamic)?\n override var oncancel: ((Event) -> dynamic)?\n override var oncanplay: ((Event) -> dynamic)?\n override var oncanplaythrough: ((Event) -> dynamic)?\n override var onchange: ((Event) -> dynamic)?\n override var onclick: ((MouseEvent) -> dynamic)?\n override var onclose: ((Event) -> dynamic)?\n override var oncontextmenu: ((MouseEvent) -> dynamic)?\n override var oncuechange: ((Event) -> dynamic)?\n override var ondblclick: ((MouseEvent) -> dynamic)?\n override var ondrag: ((DragEvent) -> dynamic)?\n override var ondragend: ((DragEvent) -> dynamic)?\n override var ondragenter: ((DragEvent) -> dynamic)?\n override var ondragexit: ((DragEvent) -> dynamic)?\n override var ondragleave: ((DragEvent) -> dynamic)?\n override var ondragover: ((DragEvent) -> dynamic)?\n override var ondragstart: ((DragEvent) -> dynamic)?\n override var ondrop: ((DragEvent) -> dynamic)?\n override var ondurationchange: ((Event) -> dynamic)?\n override var onemptied: ((Event) -> dynamic)?\n override var onended: ((Event) -> dynamic)?\n override var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?\n override var onfocus:

((FocusEvent) -> dynamic)?\n   override var oninput: ((InputEvent) -> dynamic)?\n   override var oninvalid: ((Event) -> dynamic)?\n   override var onkeydown: ((KeyboardEvent) -> dynamic)?\n   override var onkeypress: ((KeyboardEvent) -> dynamic)?\n   override var onkeyup: ((KeyboardEvent) -> dynamic)?\n   override var onload: ((Event) -> dynamic)?\n   override var onloadeddata: ((Event) -> dynamic)?\n   override var onloadedmetadata: ((Event) -> dynamic)?\n   override var onloadend: ((Event) -> dynamic)?\n   override var onloadstart: ((ProgressEvent) -> dynamic)?\n   override var onmousedown: ((MouseEvent) -> dynamic)?\n   override var onmouseenter: ((MouseEvent) -> dynamic)?\n   override var onmouseleave: ((MouseEvent) -> dynamic)?\n override var onmousemove: ((MouseEvent) -> dynamic)?\n   override var onmouseout: ((MouseEvent) -> dynamic)?\n   override var onmouseover: ((MouseEvent) -> dynamic)?\n   override var onmouseup: ((MouseEvent) -> dynamic)?\n   override var onwheel: ((WheelEvent) -> dynamic)?\n   override var onpause: ((Event) -> dynamic)?\n   override var onplay: ((Event) -> dynamic)?\n   override var onplaying: ((Event) -> dynamic)?\n override var onprogress: ((ProgressEvent) -> dynamic)?\n   override var onratechange: ((Event) -> dynamic)?\n override var onreset: ((Event) -> dynamic)?\n   override var onresize: ((Event) -> dynamic)?\n   override var onscroll: ((Event) -> dynamic)?\n   override var onseeked: ((Event) -> dynamic)?\n   override var onseeking: ((Event) -> dynamic)?\n   override var onselect: ((Event) -> dynamic)?\n   override var onshow: ((Event) -> dynamic)?\n   override var onstalled: ((Event) -> dynamic)?\n   override var onsubmit: ((Event) -> dynamic)?\n override var onsuspend: ((Event) -> dynamic)?\n   override var ontimeupdate: ((Event) -> dynamic)?\n   override var ontoggle: ((Event) -> dynamic)?\n   override var onvolumechange: ((Event) -> dynamic)?\n   override var onwaiting: ((Event) -> dynamic)?\n   override var ongotpointercapture: ((PointerEvent) -> dynamic)?\n   override var onlostpointercapture: ((PointerEvent) -> dynamic)?\n   override var onpointerdown: ((PointerEvent) -> dynamic)?\n   override var onpointermove: ((PointerEvent) -> dynamic)?\n   override var onpointerup: ((PointerEvent) -> dynamic)?\n   override var onpointercancel: ((PointerEvent) -> dynamic)?\n   override var onpointerover: ((PointerEvent) -> dynamic)?\n   override var onpointerout: ((PointerEvent) -> dynamic)?\n override var onpointerenter: ((PointerEvent) -> dynamic)?\n   override var onpointerleave: ((PointerEvent) -> dynamic)?\n   override var oncopy: ((ClipboardEvent) -> dynamic)?\n   override var oncut: ((ClipboardEvent) -> dynamic)?\n   override var onpaste: ((ClipboardEvent) -> dynamic)?\n   override var contentEditable: String\n override val isContentEditable: Boolean\n   override val style: CSSStyleDeclaration\n   override val children: HTMLCollection\n   override val firstElementChild: Element?\n   override val lastElementChild: Element?\n override val childElementCount: Int\n   override val previousElementSibling: Element?\n   override val nextElementSibling: Element?\n   override val assignedSlot: HTMLSlotElement?\n   override fun prepend(vararg nodes: dynamic)\n   override fun append(vararg nodes: dynamic)\n   override fun querySelector(selectors: String): Element?\n   override fun querySelectorAll(selectors: String): NodeList\n   override fun before(vararg nodes: dynamic)\n   override fun after(vararg nodes: dynamic)\n   override fun replaceWith(vararg nodes: dynamic)\n override fun remove()\n   override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>\n   override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n   override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint\n\n   companion object {\n   val NETWORK_EMPTY: Short\n   val NETWORK_IDLE: Short\n   val NETWORK_LOADING: Short\n   val NETWORK_NO_SOURCE: Short\n   val HAVE_NOTHING: Short\n   val HAVE_METADATA: Short\n   val HAVE_CURRENT_DATA: Short\n   val HAVE_FUTURE_DATA: Short\n   val HAVE_ENOUGH_DATA: Short\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n

```
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[Option](https://developer.mozilla.org/en/docs/Web/API/Option) to Kotlin\n */\npublic external open class
Option(text: String = definedExternally, value: String = definedExternally, defaultSelected: Boolean =
definedExternally, selected: Boolean = definedExternally) : HTMLOptionElement {\n    override var onabort:
((Event) -> dynamic)?\n    override var onblur: ((FocusEvent) -> dynamic)?\n    override var oncancel: ((Event) ->
dynamic)?\n    override var oncanplay: ((Event) -> dynamic)?\n    override var oncanplaythrough: ((Event) ->
dynamic)?\n    override var onchange: ((Event) -> dynamic)?\n    override var onclick: ((MouseEvent) ->
dynamic)?\n    override var onclose: ((Event) -> dynamic)?\n    override var oncontextmenu: ((MouseEvent) ->
dynamic)?\n    override var oncuechange: ((Event) -> dynamic)?\n    override var ondblclick: ((MouseEvent) ->
dynamic)?\n    override var ondrag: ((DragEvent) -> dynamic)?\n    override var ondragend: ((DragEvent) ->
dynamic)?\n    override var ondragenter: ((DragEvent) -> dynamic)?\n    override var ondragexit: ((DragEvent) ->
dynamic)?\n    override var ondragleave: ((DragEvent) -> dynamic)?\n    override var ondragover: ((DragEvent) ->
dynamic)?\n    override var ondragstart: ((DragEvent) -> dynamic)?\n    override var ondrop: ((DragEvent) ->
dynamic)?\n    override var ondurationchange: ((Event) -> dynamic)?\n    override var onemptied: ((Event) ->
dynamic)?\n    override var onended: ((Event) -> dynamic)?\n    override var onerror: ((dynamic, String, Int, Int,
Any?) -> dynamic)?\n    override var onfocus: ((FocusEvent) -> dynamic)?\n    override var oninput: ((InputEvent) -
> dynamic)?\n    override var oninvalid: ((Event) -> dynamic)?\n    override var onkeydown: ((KeyboardEvent) ->
dynamic)?\n    override var onkeypress: ((KeyboardEvent) -> dynamic)?\n    override var onkeyup:
((KeyboardEvent) -> dynamic)?\n    override var onload: ((Event) -> dynamic)?\n    override var onloadeddata:
((Event) -> dynamic)?\n    override var onloadedmetadata: ((Event) -> dynamic)?\n    override var onloadend:
((Event) -> dynamic)?\n    override var onloadstart: ((ProgressEvent) -> dynamic)?\n    override var onmousedown:
((MouseEvent) -> dynamic)?\n    override var onmouseenter: ((MouseEvent) -> dynamic)?\n    override var
onmouseleave: ((MouseEvent) -> dynamic)?\n    override var onmousemove: ((MouseEvent) -> dynamic)?\n
override var onmouseout: ((MouseEvent) -> dynamic)?\n    override var onmouseover: ((MouseEvent) ->
dynamic)?\n    override var onmouseup: ((MouseEvent) -> dynamic)?\n    override var onwheel: ((WheelEvent) ->
dynamic)?\n    override var onpause: ((Event) -> dynamic)?\n    override var onplay: ((Event) -> dynamic)?\n
override var onplaying: ((Event) -> dynamic)?\n    override var onprogress: ((ProgressEvent) -> dynamic)?\n
override var onratechange: ((Event) -> dynamic)?\n    override var onreset: ((Event) -> dynamic)?\n    override var
onresize: ((Event) -> dynamic)?\n    override var onscroll: ((Event) -> dynamic)?\n    override var onseeked:
((Event) -> dynamic)?\n    override var onseeking: ((Event) -> dynamic)?\n    override var onselect: ((Event) ->
dynamic)?\n    override var onshow: ((Event) -> dynamic)?\n    override var onstalled: ((Event) -> dynamic)?\n
override var onsubmit: ((Event) -> dynamic)?\n    override var onsuspend: ((Event) -> dynamic)?\n    override var
ontimeupdate: ((Event) -> dynamic)?\n    override var ontoggle: ((Event) -> dynamic)?\n    override var
onvolumechange: ((Event) -> dynamic)?\n    override var onwaiting: ((Event) -> dynamic)?\n    override var
ongotpointercapture: ((PointerEvent) -> dynamic)?\n    override var onlostpointercapture: ((PointerEvent) ->
dynamic)?\n    override var onpointerdown: ((PointerEvent) -> dynamic)?\n    override var onpointermove:
((PointerEvent) -> dynamic)?\n    override var onpointerup: ((PointerEvent) -> dynamic)?\n    override var
onpointercancel: ((PointerEvent) -> dynamic)?\n    override var onpointerover: ((PointerEvent) -> dynamic)?\n
override var onpointerout: ((PointerEvent) -> dynamic)?\n    override var onpointerenter: ((PointerEvent) ->
dynamic)?\n    override var onpointerleave: ((PointerEvent) -> dynamic)?\n    override var oncopy:
((ClipboardEvent) -> dynamic)?\n    override var oncut: ((ClipboardEvent) -> dynamic)?\n    override var onpaste:
((ClipboardEvent) -> dynamic)?\n    override var contentEditable: String\n    override val isContentEditable:
Boolean\n    override val style: CSSStyleDeclaration\n    override val children: HTMLCollection\n    override val
firstElementChild: Element?\n    override val lastElementChild: Element?\n    override val childElementCount: Int\n
  override val previousElementSibling: Element?\n    override val nextElementSibling: Element?\n    override val
assignedSlot: HTMLSlotElement?\n    override fun prepend(vararg nodes: dynamic)\n    override fun append(vararg
```

nodes: dynamic)\n   override fun querySelector(selectors: String): Element?\n   override fun querySelectorAll(selectors: String): NodeList\n   override fun before(vararg nodes: dynamic)\n   override fun after(vararg nodes: dynamic)\n   override fun replaceWith(vararg nodes: dynamic)\n   override fun remove()\n   override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>\n   override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n   override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n   override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\npublic external interface UnionElementOrHTMLCollection\n\npublic external interface UnionElementOrRadioNodeList\n\npublic external interface UnionHTMLOptGroupElementOrHTMLOptionElement\n\npublic external interface UnionAudioTrackOrTextTrackOrVideoTrack\n\npublic external interface UnionElementOrMouseEvent\n\npublic external interface UnionMessagePortOrWindowProxy\n\npublic external interface MediaProvider\n\npublic external interface RenderingContext\n\npublic external interface HTMLOrSVGImageElement : CanvasImageSource\n\npublic external interface CanvasImageSource : ImageBitmapSource\n\npublic external interface ImageBitmapSource\n\npublic external interface HTMLOrSVGScriptElement\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface DocumentReadyState {\n   companion object\n}\n\npublic inline val DocumentReadyState.Companion.LOADING: DocumentReadyState get() = \"loading\".asDynamic().unsafeCast<DocumentReadyState>()\n\npublic inline val DocumentReadyState.Companion.INTERACTIVE: DocumentReadyState get() = \"interactive\".asDynamic().unsafeCast<DocumentReadyState>()\n\npublic inline val DocumentReadyState.Companion.COMPLETE: DocumentReadyState get() = \"complete\".asDynamic().unsafeCast<DocumentReadyState>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface CanPlayTypeResult {\n   companion object\n}\n\npublic inline val CanPlayTypeResult.Companion.EMPTY: CanPlayTypeResult get() = \"\".asDynamic().unsafeCast<CanPlayTypeResult>()\n\npublic inline val CanPlayTypeResult.Companion.MAYBE: CanPlayTypeResult get() = \"maybe\".asDynamic().unsafeCast<CanPlayTypeResult>()\n\npublic inline val CanPlayTypeResult.Companion.PROBABLY: CanPlayTypeResult get() = \"probably\".asDynamic().unsafeCast<CanPlayTypeResult>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface TextTrackMode {\n   companion object\n}\n\npublic inline val TextTrackMode.Companion.DISABLED: TextTrackMode get() = \"disabled\".asDynamic().unsafeCast<TextTrackMode>()\n\npublic inline val TextTrackMode.Companion.HIDDEN: TextTrackMode get() = \"hidden\".asDynamic().unsafeCast<TextTrackMode>()\n\npublic inline val TextTrackMode.Companion.SHOWING: TextTrackMode get() = \"showing\".asDynamic().unsafeCast<TextTrackMode>()\n\n/* please, don't implement this interface!

/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface TextTrackKind {\n   companion object\n}\n\npublic inline val TextTrackKind.Companion.SUBTITLES: TextTrackKind get() = \"subtitles\".asDynamic().unsafeCast<TextTrackKind>()\n\npublic inline val TextTrackKind.Companion.CAPTIONS: TextTrackKind get() = \"captions\".asDynamic().unsafeCast<TextTrackKind>()\n\npublic inline val TextTrackKind.Companion.DESCRIPTIONS: TextTrackKind get() = \"descriptions\".asDynamic().unsafeCast<TextTrackKind>()\n\npublic inline val TextTrackKind.Companion.CHAPTERS: TextTrackKind get() = \"chapters\".asDynamic().unsafeCast<TextTrackKind>()\n\npublic inline val TextTrackKind.Companion.METADATA: TextTrackKind get() = \"metadata\".asDynamic().unsafeCast<TextTrackKind>()\n\n/ please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface SelectionMode {\n   companion object\n}\n\npublic inline val SelectionMode.Companion.SELECT: SelectionMode get() = \"select\".asDynamic().unsafeCast<SelectionMode>()\n\npublic inline val SelectionMode.Companion.START: SelectionMode get() = \"start\".asDynamic().unsafeCast<SelectionMode>()\n\npublic inline val SelectionMode.Companion.END: SelectionMode get() = \"end\".asDynamic().unsafeCast<SelectionMode>()\n\npublic inline val SelectionMode.Companion.PRESERVE: SelectionMode get() = \"preserve\".asDynamic().unsafeCast<SelectionMode>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface CanvasFillRule {\n   companion object\n}\n\npublic inline val CanvasFillRule.Companion.NONZERO: CanvasFillRule get() = \"nonzero\".asDynamic().unsafeCast<CanvasFillRule>()\n\npublic inline val CanvasFillRule.Companion.EVENODD: CanvasFillRule get() = \"evenodd\".asDynamic().unsafeCast<CanvasFillRule>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface ImageSmoothingQuality {\n   companion object\n}\n\npublic inline val ImageSmoothingQuality.Companion.LOW: ImageSmoothingQuality get() = \"low\".asDynamic().unsafeCast<ImageSmoothingQuality>()\n\npublic inline val ImageSmoothingQuality.Companion.MEDIUM: ImageSmoothingQuality get() = \"medium\".asDynamic().unsafeCast<ImageSmoothingQuality>()\n\npublic inline val ImageSmoothingQuality.Companion.HIGH: ImageSmoothingQuality get() = \"high\".asDynamic().unsafeCast<ImageSmoothingQuality>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface CanvasLineCap {\n   companion object\n}\n\npublic inline val CanvasLineCap.Companion.BUTT: CanvasLineCap get() = \"butt\".asDynamic().unsafeCast<CanvasLineCap>()\n\npublic inline val CanvasLineCap.Companion.ROUND: CanvasLineCap get() = \"round\".asDynamic().unsafeCast<CanvasLineCap>()\n\npublic inline val CanvasLineCap.Companion.SQUARE: CanvasLineCap get() = \"square\".asDynamic().unsafeCast<CanvasLineCap>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface CanvasLineJoin {\n   companion object\n}\n\npublic inline val CanvasLineJoin.Companion.ROUND: CanvasLineJoin get() = \"round\".asDynamic().unsafeCast<CanvasLineJoin>()\n\npublic inline val CanvasLineJoin.Companion.BEVEL: CanvasLineJoin get() = \"bevel\".asDynamic().unsafeCast<CanvasLineJoin>()\n\npublic inline val CanvasLineJoin.Companion.MITER: CanvasLineJoin get() = \"miter\".asDynamic().unsafeCast<CanvasLineJoin>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface CanvasTextAlign {\n   companion object\n}\n\npublic inline val CanvasTextAlign.Companion.START: CanvasTextAlign get() = \"start\".asDynamic().unsafeCast<CanvasTextAlign>()\n\npublic inline val

CanvasTextAlign.Companion.END: CanvasTextAlign get() =
\"end\".asDynamic().unsafeCast<CanvasTextAlign>()\n\npublic inline val CanvasTextAlign.Companion.LEFT:
CanvasTextAlign get() = \"left\".asDynamic().unsafeCast<CanvasTextAlign>()\n\npublic inline val
CanvasTextAlign.Companion.RIGHT: CanvasTextAlign get() =
\"right\".asDynamic().unsafeCast<CanvasTextAlign>()\n\npublic inline val
CanvasTextAlign.Companion.CENTER: CanvasTextAlign get() =
\"center\".asDynamic().unsafeCast<CanvasTextAlign>()\n\n/* please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface CanvasTextBaseline {\n   companion object\n}\n\npublic inline val CanvasTextBaseline.Companion.TOP:
CanvasTextBaseline get() = \"top\".asDynamic().unsafeCast<CanvasTextBaseline>()\n\npublic inline val
CanvasTextBaseline.Companion.HANGING: CanvasTextBaseline get() =
\"hanging\".asDynamic().unsafeCast<CanvasTextBaseline>()\n\npublic inline val
CanvasTextBaseline.Companion.MIDDLE: CanvasTextBaseline get() =
\"middle\".asDynamic().unsafeCast<CanvasTextBaseline>()\n\npublic inline val
CanvasTextBaseline.Companion.ALPHABETIC: CanvasTextBaseline get() =
\"alphabetic\".asDynamic().unsafeCast<CanvasTextBaseline>()\n\npublic inline val
CanvasTextBaseline.Companion.IDEOGRAPHIC: CanvasTextBaseline get() =
\"ideographic\".asDynamic().unsafeCast<CanvasTextBaseline>()\n\npublic inline val
CanvasTextBaseline.Companion.BOTTOM: CanvasTextBaseline get() =
\"bottom\".asDynamic().unsafeCast<CanvasTextBaseline>()\n\n/* please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface CanvasDirection {\n   companion object\n}\n\npublic inline val CanvasDirection.Companion.LTR:
CanvasDirection get() = \"ltr\".asDynamic().unsafeCast<CanvasDirection>()\n\npublic inline val
CanvasDirection.Companion.RTL: CanvasDirection get() =
\"rtl\".asDynamic().unsafeCast<CanvasDirection>()\n\npublic inline val CanvasDirection.Companion.INHERIT:
CanvasDirection get() = \"inherit\".asDynamic().unsafeCast<CanvasDirection>()\n\n/* please, don't implement this
interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic
external interface ScrollRestoration {\n   companion object\n}\n\npublic inline val
ScrollRestoration.Companion.AUTO: ScrollRestoration get() =
\"auto\".asDynamic().unsafeCast<ScrollRestoration>()\n\npublic inline val
ScrollRestoration.Companion.MANUAL: ScrollRestoration get() =
\"manual\".asDynamic().unsafeCast<ScrollRestoration>()\n\n/* please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface ImageOrientation {\n   companion object\n}\n\npublic inline val ImageOrientation.Companion.NONE:
ImageOrientation get() = \"none\".asDynamic().unsafeCast<ImageOrientation>()\n\npublic inline val
ImageOrientation.Companion.FLIPY: ImageOrientation get() =
\"flipY\".asDynamic().unsafeCast<ImageOrientation>()\n\n/* please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface PremultiplyAlpha {\n   companion object\n}\n\npublic inline val PremultiplyAlpha.Companion.NONE:
PremultiplyAlpha get() = \"none\".asDynamic().unsafeCast<PremultiplyAlpha>()\n\npublic inline val
PremultiplyAlpha.Companion.PREMULTIPLY: PremultiplyAlpha get() =
\"premultiply\".asDynamic().unsafeCast<PremultiplyAlpha>()\n\npublic inline val
PremultiplyAlpha.Companion.DEFAULT: PremultiplyAlpha get() =
\"default\".asDynamic().unsafeCast<PremultiplyAlpha>()\n\n/* please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface ColorSpaceConversion {\n   companion object\n}\n\npublic inline val
ColorSpaceConversion.Companion.NONE: ColorSpaceConversion get() =
\"none\".asDynamic().unsafeCast<ColorSpaceConversion>()\n\npublic inline val

ColorSpaceConversion.Companion.DEFAULT: ColorSpaceConversion get() =
\"default\".asDynamic().unsafeCast<ColorSpaceConversion>()\n\n/* please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface ResizeQuality {\n   companion object\n}\n\npublic inline val ResizeQuality.Companion.PIXELATED:
ResizeQuality get() = \"pixelated\".asDynamic().unsafeCast<ResizeQuality>()\n\npublic inline val
ResizeQuality.Companion.LOW: ResizeQuality get() =
\"low\".asDynamic().unsafeCast<ResizeQuality>()\n\npublic inline val ResizeQuality.Companion.MEDIUM:
ResizeQuality get() = \"medium\".asDynamic().unsafeCast<ResizeQuality>()\n\npublic inline val
ResizeQuality.Companion.HIGH: ResizeQuality get() = \"high\".asDynamic().unsafeCast<ResizeQuality>()\n\n/*
please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface BinaryType {\n   companion object\n}\n\npublic inline val BinaryType.Companion.BLOB: BinaryType
get() = \"blob\".asDynamic().unsafeCast<BinaryType>()\n\npublic inline val
BinaryType.Companion.ARRAYBUFFER: BinaryType get() =
\"arraybuffer\".asDynamic().unsafeCast<BinaryType>()\n\n/* please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface WorkerType {\n   companion object\n}\n\npublic inline val WorkerType.Companion.CLASSIC:
WorkerType get() = \"classic\".asDynamic().unsafeCast<WorkerType>()\n\npublic inline val
WorkerType.Companion.MODULE: WorkerType get() =
\"module\".asDynamic().unsafeCast<WorkerType>()\n\n/* please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface ShadowRootMode {\n   companion object\n}\n\npublic inline val ShadowRootMode.Companion.OPEN:
ShadowRootMode get() = \"open\".asDynamic().unsafeCast<ShadowRootMode>()\n\npublic inline val
ShadowRootMode.Companion.CLOSED: ShadowRootMode get() =
\"closed\".asDynamic().unsafeCast<ShadowRootMode>()\n\n/* please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface ScrollBehavior {\n   companion object\n}\n\npublic inline val ScrollBehavior.Companion.AUTO:
ScrollBehavior get() = \"auto\".asDynamic().unsafeCast<ScrollBehavior>()\n\npublic inline val
ScrollBehavior.Companion.INSTANT: ScrollBehavior get() =
\"instant\".asDynamic().unsafeCast<ScrollBehavior>()\n\npublic inline val ScrollBehavior.Companion.SMOOTH:
ScrollBehavior get() = \"smooth\".asDynamic().unsafeCast<ScrollBehavior>()\n\n/* please, don't implement this
interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic
external interface ScrollLogicalPosition {\n   companion object\n}\n\npublic inline val
ScrollLogicalPosition.Companion.START: ScrollLogicalPosition get() =
\"start\".asDynamic().unsafeCast<ScrollLogicalPosition>()\n\npublic inline val
ScrollLogicalPosition.Companion.CENTER: ScrollLogicalPosition get() =
\"center\".asDynamic().unsafeCast<ScrollLogicalPosition>()\n\npublic inline val
ScrollLogicalPosition.Companion.END: ScrollLogicalPosition get() =
\"end\".asDynamic().unsafeCast<ScrollLogicalPosition>()\n\npublic inline val
ScrollLogicalPosition.Companion.NEAREST: ScrollLogicalPosition get() =
\"nearest\".asDynamic().unsafeCast<ScrollLogicalPosition>()\n\n/* please, don't implement this interface!
*/\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external
interface CSSBoxType {\n   companion object\n}\n\npublic inline val CSSBoxType.Companion.MARGIN:
CSSBoxType get() = \"margin\".asDynamic().unsafeCast<CSSBoxType>()\n\npublic inline val
CSSBoxType.Companion.BORDER: CSSBoxType get() =
\"border\".asDynamic().unsafeCast<CSSBoxType>()\n\npublic inline val CSSBoxType.Companion.PADDING:
CSSBoxType get() = \"padding\".asDynamic().unsafeCast<CSSBoxType>()\n\npublic inline val
CSSBoxType.Companion.CONTENT: CSSBoxType get() =

\"content\".asDynamic().unsafeCast<CSSBoxType>()","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.fetch\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.files.*\nimport org.w3c.xhr.*\n\n/**\n * Exposes the JavaScript [Headers](https://developer.mozilla.org/en/docs/Web/API/Headers) to Kotlin\n */\npublic external open class Headers(init: dynamic = definedExternally) {\n   fun append(name: String, value: String)\n   fun delete(name: String)\n   fun get(name: String): String?\n   fun has(name: String): Boolean\n   fun set(name: String, value: String)\n}\n\n/**\n * Exposes the JavaScript [Body](https://developer.mozilla.org/en/docs/Web/API/Body) to Kotlin\n */\npublic external interface Body {\n   val bodyUsed: Boolean\n   fun arrayBuffer(): Promise<ArrayBuffer>\n   fun blob(): Promise<Blob>\n   fun formData(): Promise<FormData>\n   fun json(): Promise<Any?>\n   fun text(): Promise<String>\n}\n\n/**\n * Exposes the JavaScript [Request](https://developer.mozilla.org/en/docs/Web/API/Request) to Kotlin\n */\npublic external open class Request(input: dynamic, init: RequestInit = definedExternally) : Body {\n   open val method: String\n   open val url: String\n   open val headers: Headers\n   open val type: RequestType\n   open val destination: RequestDestination\n   open val referrer: String\n   open val referrerPolicy: dynamic\n   open val mode: RequestMode\n   open val credentials: RequestCredentials\n   open val cache: RequestCache\n   open val redirect: RequestRedirect\n   open val integrity: String\n   open val keepalive: Boolean\n   override val bodyUsed: Boolean\n   fun clone(): Request\n   override fun arrayBuffer(): Promise<ArrayBuffer>\n   override fun blob(): Promise<Blob>\n   override fun formData(): Promise<FormData>\n   override fun json(): Promise<Any?>\n   override fun text(): Promise<String>\n}\n\npublic external interface RequestInit {\n   var method: String?\n   get() = definedExternally\n   set(value) = definedExternally\n   var headers: dynamic\n   get() = definedExternally\n   set(value) = definedExternally\n   var body: dynamic\n   get() = definedExternally\n   set(value) = definedExternally\n   var referrer: String?\n   get() = definedExternally\n   set(value) = definedExternally\n   var referrerPolicy: dynamic\n   get() = definedExternally\n   set(value) = definedExternally\n   var mode: RequestMode?\n   get() = definedExternally\n   set(value) = definedExternally\n   var credentials: RequestCredentials?\n   get() = definedExternally\n   set(value) = definedExternally\n   var cache: RequestCache?\n   get() = definedExternally\n   set(value) = definedExternally\n   var redirect: RequestRedirect?\n   get() = definedExternally\n   set(value) = definedExternally\n   var integrity: String?\n   get() = definedExternally\n   set(value) = definedExternally\n   var keepalive: Boolean?\n   get() = definedExternally\n   set(value) = definedExternally\n   var window: Any?\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun RequestInit(method: String? = undefined, headers: dynamic = undefined, body: dynamic = undefined, referrer: String? = undefined, referrerPolicy: dynamic = undefined, mode: RequestMode? = undefined, credentials: RequestCredentials? = undefined, cache: RequestCache? = undefined, redirect: RequestRedirect? = undefined, integrity: String? = undefined, keepalive: Boolean? = undefined, window: Any? = undefined): RequestInit {\n   val o = js(\"({})\")\n   o[\"method\"] = method\n   o[\"headers\"] = headers\n   o[\"body\"] = body\n   o[\"referrer\"] = referrer\n   o[\"referrerPolicy\"] = referrerPolicy\n   o[\"mode\"] = mode\n   o[\"credentials\"] = credentials\n   o[\"cache\"] = cache\n   o[\"redirect\"] = redirect\n   o[\"integrity\"] = integrity\n   o[\"keepalive\"] = keepalive\n   o[\"window\"] = window\n   return o\n}\n\n/**\n * Exposes the JavaScript [Response](https://developer.mozilla.org/en/docs/Web/API/Response) to Kotlin\n */\npublic external open class Response(body: dynamic = definedExternally, init: ResponseInit = definedExternally) : Body {\n   open val type: ResponseType\n   open val url: String\n   open val redirected: Boolean\n   open val status: Short\n   open val ok: Boolean\n   open val statusText: String\n   open val headers: Headers\n   open val body: dynamic\n   open val trailer: Promise<Headers>\n   override val bodyUsed: Boolean\n   fun clone(): Response\n   override fun arrayBuffer(): Promise<ArrayBuffer>\n   override fun blob(): Promise<Blob>\n   override fun formData(): Promise<FormData>\n   override fun json(): Promise<Any?>\n

override fun text(): Promise<String>\n\n   companion object {\n   fun error(): Response\n   fun redirect(url: String, status: Short = definedExternally): Response\n   }\n}\n\npublic external interface ResponseInit {\n   var status: Short? /* = 200 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var statusText: String? /* = \"OK\" */\n   get() = definedExternally\n   set(value) = definedExternally\n   var headers: dynamic\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ResponseInit(status: Short? = 200, statusText: String? = \"OK\", headers: dynamic = undefined): ResponseInit {\n   val o = js(\"({})\")\n   o[\"status\"] = status\n   o[\"statusText\"] = statusText\n   o[\"headers\"] = headers\n   return o\n}\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface RequestType {\n   companion object\n}\n\npublic inline val RequestType.Companion.EMPTY: RequestType get() = \"\".asDynamic().unsafeCast<RequestType>()\n\npublic inline val RequestType.Companion.AUDIO: RequestType get() = \"audio\".asDynamic().unsafeCast<RequestType>()\n\npublic inline val RequestType.Companion.FONT: RequestType get() = \"font\".asDynamic().unsafeCast<RequestType>()\n\npublic inline val RequestType.Companion.IMAGE: RequestType get() = \"image\".asDynamic().unsafeCast<RequestType>()\n\npublic inline val RequestType.Companion.SCRIPT: RequestType get() = \"script\".asDynamic().unsafeCast<RequestType>()\n\npublic inline val RequestType.Companion.STYLE: RequestType get() = \"style\".asDynamic().unsafeCast<RequestType>()\n\npublic inline val RequestType.Companion.TRACK: RequestType get() = \"track\".asDynamic().unsafeCast<RequestType>()\n\npublic inline val RequestType.Companion.VIDEO: RequestType get() = \"video\".asDynamic().unsafeCast<RequestType>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface RequestDestination {\n   companion object\n}\n\npublic inline val RequestDestination.Companion.EMPTY: RequestDestination get() = \"\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.DOCUMENT: RequestDestination get() = \"document\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.EMBED: RequestDestination get() = \"embed\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.FONT: RequestDestination get() = \"font\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.IMAGE: RequestDestination get() = \"image\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.MANIFEST: RequestDestination get() = \"manifest\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.MEDIA: RequestDestination get() = \"media\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.OBJECT: RequestDestination get() = \"object\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.REPORT: RequestDestination get() = \"report\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.SCRIPT: RequestDestination get() = \"script\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.SERVICEWORKER: RequestDestination get() = \"serviceworker\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.SHAREDWORKER: RequestDestination get() =

\"sharedworker\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.STYLE: RequestDestination get() = \"style\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.WORKER: RequestDestination get() = \"worker\".asDynamic().unsafeCast<RequestDestination>()\n\npublic inline val RequestDestination.Companion.XSLT: RequestDestination get() = \"xslt\".asDynamic().unsafeCast<RequestDestination>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface RequestMode {\n   companion object\n}\n\npublic inline val RequestMode.Companion.NAVIGATE: RequestMode get() = \"navigate\".asDynamic().unsafeCast<RequestMode>()\n\npublic inline val RequestMode.Companion.SAME_ORIGIN: RequestMode get() = \"same-origin\".asDynamic().unsafeCast<RequestMode>()\n\npublic inline val RequestMode.Companion.NO_CORS: RequestMode get() = \"no-cors\".asDynamic().unsafeCast<RequestMode>()\n\npublic inline val RequestMode.Companion.CORS: RequestMode get() = \"cors\".asDynamic().unsafeCast<RequestMode>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface RequestCredentials {\n   companion object\n}\n\npublic inline val RequestCredentials.Companion.OMIT: RequestCredentials get() = \"omit\".asDynamic().unsafeCast<RequestCredentials>()\n\npublic inline val RequestCredentials.Companion.SAME_ORIGIN: RequestCredentials get() = \"same-origin\".asDynamic().unsafeCast<RequestCredentials>()\n\npublic inline val RequestCredentials.Companion.INCLUDE: RequestCredentials get() = \"include\".asDynamic().unsafeCast<RequestCredentials>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface RequestCache {\n   companion object\n}\n\npublic inline val RequestCache.Companion.DEFAULT: RequestCache get() = \"default\".asDynamic().unsafeCast<RequestCache>()\n\npublic inline val RequestCache.Companion.NO_STORE: RequestCache get() = \"no-store\".asDynamic().unsafeCast<RequestCache>()\n\npublic inline val RequestCache.Companion.RELOAD: RequestCache get() = \"reload\".asDynamic().unsafeCast<RequestCache>()\n\npublic inline val RequestCache.Companion.NO_CACHE: RequestCache get() = \"no-cache\".asDynamic().unsafeCast<RequestCache>()\n\npublic inline val RequestCache.Companion.FORCE_CACHE: RequestCache get() = \"force-cache\".asDynamic().unsafeCast<RequestCache>()\n\npublic inline val RequestCache.Companion.ONLY_IF_CACHED: RequestCache get() = \"only-if-cached\".asDynamic().unsafeCast<RequestCache>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface RequestRedirect {\n   companion object\n}\n\npublic inline val RequestRedirect.Companion.FOLLOW: RequestRedirect get() = \"follow\".asDynamic().unsafeCast<RequestRedirect>()\n\npublic inline val RequestRedirect.Companion.ERROR: RequestRedirect get() = \"error\".asDynamic().unsafeCast<RequestRedirect>()\n\npublic inline val RequestRedirect.Companion.MANUAL: RequestRedirect get() = \"manual\".asDynamic().unsafeCast<RequestRedirect>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface ResponseType {\n   companion object\n}\n\npublic inline val ResponseType.Companion.BASIC: ResponseType get() = \"basic\".asDynamic().unsafeCast<ResponseType>()\n\npublic inline val ResponseType.Companion.CORS: ResponseType get() = \"cors\".asDynamic().unsafeCast<ResponseType>()\n\npublic inline val ResponseType.Companion.DEFAULT: ResponseType get() = \"default\".asDynamic().unsafeCast<ResponseType>()\n\npublic inline val ResponseType.Companion.ERROR: ResponseType get() = \"error\".asDynamic().unsafeCast<ResponseType>()\n\npublic inline val ResponseType.Companion.OPAQUE:

ResponseType get() = \"opaque\".asDynamic().unsafeCast<ResponseType>()\n\npublic inline val ResponseType.Companion.OPAQUEREDIRECT: ResponseType get() = \"opaqueredirect\".asDynamic().unsafeCast<ResponseType>()","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.dom.mediacapture\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\nimport org.w3c.dom.events.*\n\n/**\n * Exposes the JavaScript [MediaStream](https://developer.mozilla.org/en/docs/Web/API/MediaStream) to Kotlin\n */\npublic external open class MediaStream() : EventTarget, MediaProvider {\n   constructor(stream: MediaStream)\n   constructor(tracks: Array<MediaStreamTrack>)\n   open val id: String\n   open val active: Boolean\n   var onaddtrack: ((MediaStreamTrackEvent) -> dynamic)?\n   var onremovetrack: ((MediaStreamTrackEvent) -> dynamic)?\n   fun getAudioTracks(): Array<MediaStreamTrack>\n   fun getVideoTracks(): Array<MediaStreamTrack>\n   fun getTracks(): Array<MediaStreamTrack>\n   fun getTrackById(trackId: String): MediaStreamTrack?\n   fun addTrack(track: MediaStreamTrack)\n   fun removeTrack(track: MediaStreamTrack)\n   fun clone(): MediaStream\n}\n\n/**\n * Exposes the JavaScript [MediaStreamTrack](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrack) to Kotlin\n */\npublic external abstract class MediaStreamTrack : EventTarget {\n   open val kind: String\n   open val id: String\n   open val label: String\n   open var enabled: Boolean\n   open val muted: Boolean\n   open var onmute: ((Event) -> dynamic)?\n   open var onunmute: ((Event) -> dynamic)?\n   open val readyState: MediaStreamTrackState\n   open var onended: ((Event) -> dynamic)?\n   open var onoverconstrained: ((Event) -> dynamic)?\n   fun clone(): MediaStreamTrack\n   fun stop()\n   fun getCapabilities(): MediaTrackCapabilities\n   fun getConstraints(): MediaTrackConstraints\n   fun getSettings(): MediaTrackSettings\n   fun applyConstraints(constraints: MediaTrackConstraints = definedExternally): Promise<Unit>\n}\n\n/**\n * Exposes the JavaScript [MediaTrackSupportedConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSupportedConstraints) to Kotlin\n */\npublic external interface MediaTrackSupportedConstraints {\n   var width: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var height: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var aspectRatio: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var frameRate: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var facingMode: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var resizeMode: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var volume: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var sampleRate: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var sampleSize: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var echoCancellation: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var autoGainControl: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var noiseSuppression: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var latency: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var channelCount: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var deviceId: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n   var groupId: Boolean? /* = true */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaTrackSupportedConstraints(width: Boolean? = true, height: Boolean? = true, aspectRatio: Boolean? = true, frameRate: Boolean? = true, facingMode: Boolean? = true, resizeMode: Boolean? = true, volume: Boolean? = true, sampleRate: Boolean? = true, sampleSize: Boolean? = true, echoCancellation: Boolean? = true, autoGainControl: Boolean? = true, noiseSuppression: Boolean? = true, latency: Boolean? = true, channelCount: Boolean? = true, deviceId: Boolean? = true, groupId: Boolean? = true): MediaTrackSupportedConstraints {\n   val o = js(\"({})\")\n   o[\"width\"] = width\n   o[\"height\"] = height\n   o[\"aspectRatio\"] = aspectRatio\n   o[\"frameRate\"] =

```
frameRate\n    o[\"facingMode\"] = facingMode\n    o[\"resizeMode\"] = resizeMode\n    o[\"volume\"] = volume\n
o[\"sampleRate\"] = sampleRate\n    o[\"sampleSize\"] = sampleSize\n    o[\"echoCancellation\"] =
echoCancellation\n    o[\"autoGainControl\"] = autoGainControl\n    o[\"noiseSuppression\"] = noiseSuppression\n
o[\"latency\"] = latency\n    o[\"channelCount\"] = channelCount\n    o[\"deviceId\"] = deviceId\n    o[\"groupId\"] =
groupId\n    return o\n}\n\npublic external interface MediaTrackCapabilities {\n    var width: ULongRange?\n
get() = definedExternally\n        set(value) = definedExternally\n    var height: ULongRange?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var aspectRatio: DoubleRange?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var frameRate: DoubleRange?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var facingMode: Array<String>?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var resizeMode: Array<String>?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var volume: DoubleRange?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var sampleRate: ULongRange?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var sampleSize: ULongRange?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var echoCancellation: Array<Boolean>?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var autoGainControl: Array<Boolean>?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var noiseSuppression: Array<Boolean>?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var latency: DoubleRange?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var channelCount: ULongRange?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var deviceId: String?\n        get() = definedExternally\n
  set(value) = definedExternally\n    var groupId: String?\n        get() = definedExternally\n        set(value) =
definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaTrackCapabilities(width:
ULongRange? = undefined, height: ULongRange? = undefined, aspectRatio: DoubleRange? = undefined,
frameRate: DoubleRange? = undefined, facingMode: Array<String>? = undefined, resizeMode: Array<String>? =
undefined, volume: DoubleRange? = undefined, sampleRate: ULongRange? = undefined, sampleSize:
ULongRange? = undefined, echoCancellation: Array<Boolean>? = undefined, autoGainControl: Array<Boolean>?
= undefined, noiseSuppression: Array<Boolean>? = undefined, latency: DoubleRange? = undefined, channelCount:
ULongRange? = undefined, deviceId: String? = undefined, groupId: String? = undefined): MediaTrackCapabilities
{\n    val o = js(\"({})\")\n    o[\"width\"] = width\n    o[\"height\"] = height\n    o[\"aspectRatio\"] = aspectRatio\n
o[\"frameRate\"] = frameRate\n    o[\"facingMode\"] = facingMode\n    o[\"resizeMode\"] = resizeMode\n
o[\"volume\"] = volume\n    o[\"sampleRate\"] = sampleRate\n    o[\"sampleSize\"] = sampleSize\n
o[\"echoCancellation\"] = echoCancellation\n    o[\"autoGainControl\"] = autoGainControl\n
o[\"noiseSuppression\"] = noiseSuppression\n    o[\"latency\"] = latency\n    o[\"channelCount\"] = channelCount\n
o[\"deviceId\"] = deviceId\n    o[\"groupId\"] = groupId\n    return o\n}\n\n/**\n * Exposes the JavaScript
[MediaTrackConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackConstraints) to Kotlin\n
*/\npublic external interface MediaTrackConstraints : MediaTrackConstraintSet {\n    var advanced:
Array<MediaTrackConstraintSet>?\n        get() = definedExternally\n        set(value) =
definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaTrackConstraints(advanced:
Array<MediaTrackConstraintSet>? = undefined, width: dynamic = undefined, height: dynamic = undefined,
aspectRatio: dynamic = undefined, frameRate: dynamic = undefined, facingMode: dynamic = undefined,
resizeMode: dynamic = undefined, volume: dynamic = undefined, sampleRate: dynamic = undefined, sampleSize:
dynamic = undefined, echoCancellation: dynamic = undefined, autoGainControl: dynamic = undefined,
noiseSuppression: dynamic = undefined, latency: dynamic = undefined, channelCount: dynamic = undefined,
deviceId: dynamic = undefined, groupId: dynamic = undefined): MediaTrackConstraints {\n    val o = js(\"({})\")\n
o[\"advanced\"] = advanced\n    o[\"width\"] = width\n    o[\"height\"] = height\n    o[\"aspectRatio\"] =
aspectRatio\n    o[\"frameRate\"] = frameRate\n    o[\"facingMode\"] = facingMode\n    o[\"resizeMode\"] =
resizeMode\n    o[\"volume\"] = volume\n    o[\"sampleRate\"] = sampleRate\n    o[\"sampleSize\"] = sampleSize\n
```

```
o[\"echoCancellation\"] = echoCancellation\n    o[\"autoGainControl\"] = autoGainControl\n
o[\"noiseSuppression\"] = noiseSuppression\n    o[\"latency\"] = latency\n    o[\"channelCount\"] = channelCount\n
o[\"deviceId\"] = deviceId\n    o[\"groupId\"] = groupId\n    return o\n}\n\npublic external interface
MediaTrackConstraintSet {\n    var width: dynamic\n        get() = definedExternally\n        set(value) =
definedExternally\n    var height: dynamic\n        get() = definedExternally\n        set(value) = definedExternally\n
var aspectRatio: dynamic\n        get() = definedExternally\n        set(value) = definedExternally\n    var frameRate:
dynamic\n        get() = definedExternally\n        set(value) = definedExternally\n    var facingMode: dynamic\n
get() = definedExternally\n        set(value) = definedExternally\n    var resizeMode: dynamic\n        get() =
definedExternally\n        set(value) = definedExternally\n    var volume: dynamic\n        get() = definedExternally\n
    set(value) = definedExternally\n    var sampleRate: dynamic\n        get() = definedExternally\n        set(value) =
definedExternally\n    var sampleSize: dynamic\n        get() = definedExternally\n        set(value) =
definedExternally\n    var echoCancellation: dynamic\n        get() = definedExternally\n        set(value) =
definedExternally\n    var autoGainControl: dynamic\n        get() = definedExternally\n        set(value) =
definedExternally\n    var noiseSuppression: dynamic\n        get() = definedExternally\n        set(value) =
definedExternally\n    var latency: dynamic\n        get() = definedExternally\n        set(value) = definedExternally\n
var channelCount: dynamic\n        get() = definedExternally\n        set(value) = definedExternally\n    var deviceId:
dynamic\n        get() = definedExternally\n        set(value) = definedExternally\n    var groupId: dynamic\n        get()
= definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaTrackConstraintSet(width:
dynamic = undefined, height: dynamic = undefined, aspectRatio: dynamic = undefined, frameRate: dynamic =
undefined, facingMode: dynamic = undefined, resizeMode: dynamic = undefined, volume: dynamic = undefined,
sampleRate: dynamic = undefined, sampleSize: dynamic = undefined, echoCancellation: dynamic = undefined,
autoGainControl: dynamic = undefined, noiseSuppression: dynamic = undefined, latency: dynamic = undefined,
channelCount: dynamic = undefined, deviceId: dynamic = undefined, groupId: dynamic = undefined):
MediaTrackConstraintSet {\n    val o = js(\"({})\")\n    o[\"width\"] = width\n    o[\"height\"] = height\n
o[\"aspectRatio\"] = aspectRatio\n    o[\"frameRate\"] = frameRate\n    o[\"facingMode\"] = facingMode\n
o[\"resizeMode\"] = resizeMode\n    o[\"volume\"] = volume\n    o[\"sampleRate\"] = sampleRate\n
o[\"sampleSize\"] = sampleSize\n    o[\"echoCancellation\"] = echoCancellation\n    o[\"autoGainControl\"] =
autoGainControl\n    o[\"noiseSuppression\"] = noiseSuppression\n    o[\"latency\"] = latency\n
o[\"channelCount\"] = channelCount\n    o[\"deviceId\"] = deviceId\n    o[\"groupId\"] = groupId\n    return
o\n}\n\n/**\n * Exposes the JavaScript
[MediaTrackSettings](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSettings) to Kotlin\n */\npublic
external interface MediaTrackSettings {\n    var width: Int?\n        get() = definedExternally\n        set(value) =
definedExternally\n    var height: Int?\n        get() = definedExternally\n        set(value) = definedExternally\n    var
aspectRatio: Double?\n        get() = definedExternally\n        set(value) = definedExternally\n    var frameRate:
Double?\n        get() = definedExternally\n        set(value) = definedExternally\n    var facingMode: String?\n
get() = definedExternally\n        set(value) = definedExternally\n    var resizeMode: String?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var volume: Double?\n        get() = definedExternally\n
    set(value) = definedExternally\n    var sampleRate: Int?\n        get() = definedExternally\n        set(value) =
definedExternally\n    var sampleSize: Int?\n        get() = definedExternally\n        set(value) = definedExternally\n
var echoCancellation: Boolean?\n        get() = definedExternally\n        set(value) = definedExternally\n    var
autoGainControl: Boolean?\n        get() = definedExternally\n        set(value) = definedExternally\n    var
noiseSuppression: Boolean?\n        get() = definedExternally\n        set(value) = definedExternally\n    var latency:
Double?\n        get() = definedExternally\n        set(value) = definedExternally\n    var channelCount: Int?\n
get() = definedExternally\n        set(value) = definedExternally\n    var deviceId: String?\n        get() =
definedExternally\n        set(value) = definedExternally\n    var groupId: String?\n        get() = definedExternally\n
    set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaTrackSettings(width: Int? =
```

undefined, height: Int? = undefined, aspectRatio: Double? = undefined, frameRate: Double? = undefined, facingMode: String? = undefined, resizeMode: String? = undefined, volume: Double? = undefined, sampleRate: Int? = undefined, sampleSize: Int? = undefined, echoCancellation: Boolean? = undefined, autoGainControl: Boolean? = undefined, noiseSuppression: Boolean? = undefined, latency: Double? = undefined, channelCount: Int? = undefined, deviceId: String? = undefined, groupId: String? = undefined): MediaTrackSettings {\n val o = js(\"({})\")\n o[\"width\"] = width\n o[\"height\"] = height\n o[\"aspectRatio\"] = aspectRatio\n o[\"frameRate\"] = frameRate\n o[\"facingMode\"] = facingMode\n o[\"resizeMode\"] = resizeMode\n o[\"volume\"] = volume\n o[\"sampleRate\"] = sampleRate\n o[\"sampleSize\"] = sampleSize\n o[\"echoCancellation\"] = echoCancellation\n o[\"autoGainControl\"] = autoGainControl\n o[\"noiseSuppression\"] = noiseSuppression\n o[\"latency\"] = latency\n o[\"channelCount\"] = channelCount\n o[\"deviceId\"] = deviceId\n o[\"groupId\"] = groupId\n return o\n}\n\n/**\n * Exposes the JavaScript [MediaStreamTrackEvent](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrackEvent) to Kotlin\n */\npublic external open class MediaStreamTrackEvent(type: String, eventInitDict: MediaStreamTrackEventInit) : Event {\n open val track: MediaStreamTrack\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface MediaStreamTrackEventInit : EventInit {\n var track: MediaStreamTrack?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaStreamTrackEventInit(track: MediaStreamTrack?, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): MediaStreamTrackEventInit {\n val o = js(\"({})\")\n o[\"track\"] = track\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\npublic external open class OverconstrainedErrorEvent(type: String, eventInitDict: OverconstrainedErrorEventInit) : Event {\n open val error: dynamic\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface OverconstrainedErrorEventInit : EventInit {\n var error: dynamic /* = null */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun OverconstrainedErrorEventInit(error: dynamic = null, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): OverconstrainedErrorEventInit {\n val o = js(\"({})\")\n o[\"error\"] = error\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [MediaDevices](https://developer.mozilla.org/en/docs/Web/API/MediaDevices) to Kotlin\n */\npublic external abstract class MediaDevices : EventTarget {\n open var ondevicechange: ((Event) -> dynamic)?\n fun enumerateDevices(): Promise<Array<MediaDeviceInfo>>\n fun getSupportedConstraints(): MediaTrackSupportedConstraints\n fun getUserMedia(constraints: MediaStreamConstraints = definedExternally): Promise<MediaStream>\n}\n\n/**\n * Exposes the JavaScript [MediaDeviceInfo](https://developer.mozilla.org/en/docs/Web/API/MediaDeviceInfo) to Kotlin\n */\npublic external abstract class MediaDeviceInfo {\n open val deviceId: String\n open val kind: MediaDeviceKind\n open val label: String\n open val groupId: String\n fun toJSON(): dynamic\n}\n\npublic external abstract class InputDeviceInfo : MediaDeviceInfo {\n fun getCapabilities(): MediaTrackCapabilities\n}\n\n/**\n * Exposes the JavaScript [MediaStreamConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaStreamConstraints) to Kotlin\n */\npublic external interface MediaStreamConstraints {\n var video: dynamic /* = false */\n get() = definedExternally\n set(value) = definedExternally\n var audio: dynamic /* = false */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun MediaStreamConstraints(video: dynamic = false, audio: dynamic = false): MediaStreamConstraints {\n val o = js(\"({})\")\n o[\"video\"] = video\n o[\"audio\"] = audio\n return o\n}\n\npublic external interface ConstrainablePattern {\n var onoverconstrained: ((Event) -> dynamic)?\n get() = definedExternally\n set(value) = definedExternally\n fun getCapabilities(): Capabilities\n fun getConstraints(): Constraints\n fun getSettings(): Settings\n fun

applyConstraints(constraints: Constraints = definedExternally): Promise<Unit>\n}\n\n/**\n * Exposes the JavaScript [DoubleRange](https://developer.mozilla.org/en/docs/Web/API/DoubleRange) to Kotlin\n */\npublic external interface DoubleRange {\n var max: Double?\n get() = definedExternally\n set(value) = definedExternally\n var min: Double?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleRange(max: Double? = undefined, min: Double? = undefined): DoubleRange {\n val o = js(\"({})\")\n o[\"max\"] = max\n o[\"min\"] = min\n return o\n}\n\npublic external interface ConstrainDoubleRange : DoubleRange {\n var exact: Double?\n get() = definedExternally\n set(value) = definedExternally\n var ideal: Double?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ConstrainDoubleRange(exact: Double? = undefined, ideal: Double? = undefined, max: Double? = undefined, min: Double? = undefined): ConstrainDoubleRange {\n val o = js(\"({})\")\n o[\"exact\"] = exact\n o[\"ideal\"] = ideal\n o[\"max\"] = max\n o[\"min\"] = min\n return o\n}\n\npublic external interface ULongRange {\n var max: Int?\n get() = definedExternally\n set(value) = definedExternally\n var min: Int?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ULongRange(max: Int? = undefined, min: Int? = undefined): ULongRange {\n val o = js(\"({})\")\n o[\"max\"] = max\n o[\"min\"] = min\n return o\n}\n\npublic external interface ConstrainULongRange : ULongRange {\n var exact: Int?\n get() = definedExternally\n set(value) = definedExternally\n var ideal: Int?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ConstrainULongRange(exact: Int? = undefined, ideal: Int? = undefined, max: Int? = undefined, min: Int? = undefined): ConstrainULongRange {\n val o = js(\"({})\")\n o[\"exact\"] = exact\n o[\"ideal\"] = ideal\n o[\"max\"] = max\n o[\"min\"] = min\n return o\n}\n\n/**\n * Exposes the JavaScript [ConstrainBooleanParameters](https://developer.mozilla.org/en/docs/Web/API/ConstrainBooleanParameters) to Kotlin\n */\npublic external interface ConstrainBooleanParameters {\n var exact: Boolean?\n get() = definedExternally\n set(value) = definedExternally\n var ideal: Boolean?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ConstrainBooleanParameters(exact: Boolean? = undefined, ideal: Boolean? = undefined): ConstrainBooleanParameters {\n val o = js(\"({})\")\n o[\"exact\"] = exact\n o[\"ideal\"] = ideal\n return o\n}\n\n/**\n * Exposes the JavaScript [ConstrainDOMStringParameters](https://developer.mozilla.org/en/docs/Web/API/ConstrainDOMStringParameters) to Kotlin\n */\npublic external interface ConstrainDOMStringParameters {\n var exact: dynamic\n get() = definedExternally\n set(value) = definedExternally\n var ideal: dynamic\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ConstrainDOMStringParameters(exact: dynamic = undefined, ideal: dynamic = undefined): ConstrainDOMStringParameters {\n val o = js(\"({})\")\n o[\"exact\"] = exact\n o[\"ideal\"] = ideal\n return o\n}\n\npublic external interface Capabilities\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun Capabilities(): Capabilities {\n val o = js(\"({})\")\n return o\n}\n\npublic external interface Settings\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun Settings(): Settings {\n val o = js(\"({})\")\n return o\n}\n\npublic external interface ConstraintSet\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ConstraintSet(): ConstraintSet {\n val o = js(\"({})\")\n return o\n}\n\npublic external interface Constraints : ConstraintSet {\n var advanced: Array<ConstraintSet>?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",

\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun Constraints(advanced: Array<ConstraintSet>? = undefined): Constraints {\n   val o = js(\"({})\")\n   o[\"advanced\"] = advanced\n   return o\n}\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface MediaStreamTrackState {\n   companion object\n}\n\npublic inline val MediaStreamTrackState.Companion.LIVE: MediaStreamTrackState get() = \"live\".asDynamic().unsafeCast<MediaStreamTrackState>()\n\npublic inline val MediaStreamTrackState.Companion.ENDED: MediaStreamTrackState get() = \"ended\".asDynamic().unsafeCast<MediaStreamTrackState>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface VideoFacingModeEnum {\n   companion object\n}\n\npublic inline val VideoFacingModeEnum.Companion.USER: VideoFacingModeEnum get() = \"user\".asDynamic().unsafeCast<VideoFacingModeEnum>()\n\npublic inline val VideoFacingModeEnum.Companion.ENVIRONMENT: VideoFacingModeEnum get() = \"environment\".asDynamic().unsafeCast<VideoFacingModeEnum>()\n\npublic inline val VideoFacingModeEnum.Companion.LEFT: VideoFacingModeEnum get() = \"left\".asDynamic().unsafeCast<VideoFacingModeEnum>()\n\npublic inline val VideoFacingModeEnum.Companion.RIGHT: VideoFacingModeEnum get() = \"right\".asDynamic().unsafeCast<VideoFacingModeEnum>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface VideoResizeModeEnum {\n   companion object\n}\n\npublic inline val VideoResizeModeEnum.Companion.NONE: VideoResizeModeEnum get() = \"none\".asDynamic().unsafeCast<VideoResizeModeEnum>()\n\npublic inline val VideoResizeModeEnum.Companion.CROP_AND_SCALE: VideoResizeModeEnum get() = \"crop-and-scale\".asDynamic().unsafeCast<VideoResizeModeEnum>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface MediaDeviceKind {\n   companion object\n}\n\npublic inline val MediaDeviceKind.Companion.AUDIOINPUT: MediaDeviceKind get() = \"audioinput\".asDynamic().unsafeCast<MediaDeviceKind>()\n\npublic inline val MediaDeviceKind.Companion.AUDIOOUTPUT: MediaDeviceKind get() = \"audiooutput\".asDynamic().unsafeCast<MediaDeviceKind>()\n\npublic inline val MediaDeviceKind.Companion.VIDEOINPUT: MediaDeviceKind get() = \"videoinput\".asDynamic().unsafeCast<MediaDeviceKind>()","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.dom.mediasource\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\nimport org.w3c.dom.events.*\n\n/**\n * Exposes the JavaScript [MediaSource](https://developer.mozilla.org/en/docs/Web/API/MediaSource) to Kotlin\n */\npublic external open class MediaSource : EventTarget, MediaProvider {\n   open val sourceBuffers: SourceBufferList\n   open val activeSourceBuffers: SourceBufferList\n   open val readyState: ReadyState\n   var duration: Double\n   var onsourceopen: ((Event) -> dynamic)?\n   var onsourceended: ((Event) -> dynamic)?\n   var onsourceclose: ((Event) -> dynamic)?\n   fun addSourceBuffer(type: String): SourceBuffer\n   fun removeSourceBuffer(sourceBuffer: SourceBuffer)\n   fun endOfStream(error: EndOfStreamError = definedExternally)\n   fun setLiveSeekableRange(start: Double, end: Double)\n   fun clearLiveSeekableRange()\n\n   companion object {\n   fun isTypeSupported(type: String): Boolean\n   }\n}\n\n/**\n * Exposes the JavaScript [SourceBuffer](https://developer.mozilla.org/en/docs/Web/API/SourceBuffer) to Kotlin\n */\npublic external abstract class SourceBuffer : EventTarget {\n   open var mode: AppendMode\n   open val updating: Boolean\n   open val buffered: TimeRanges\n   open var timestampOffset: Double\n   open val audioTracks: AudioTrackList\n

open val videoTracks: VideoTrackList\n   open val textTracks: TextTrackList\n   open var appendWindowStart: Double\n   open var appendWindowEnd: Double\n   open var onupdatestart: ((Event) -> dynamic)?\n   open var onupdate: ((Event) -> dynamic)?\n   open var onupdateend: ((Event) -> dynamic)?\n   open var onerror: ((Event) -> dynamic)?\n   open var onabort: ((Event) -> dynamic)?\n   fun appendBuffer(data: dynamic)\n   fun abort()\n   fun remove(start: Double, end: Double)\n}\n\n/**\n * Exposes the JavaScript [SourceBufferList](https://developer.mozilla.org/en/docs/Web/API/SourceBufferList) to Kotlin\n */\npublic external abstract class SourceBufferList : EventTarget {\n   open val length: Int\n   open var onaddsourcebuffer: ((Event) -> dynamic)?\n   open var onremovesourcebuffer: ((Event) -> dynamic)?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SourceBufferList.get(index: Int): SourceBuffer? = asDynamic()[index]\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface ReadyState {\n   companion object\n}\n\npublic inline val ReadyState.Companion.CLOSED: ReadyState get() = \"closed\".asDynamic().unsafeCast<ReadyState>()\n\npublic inline val ReadyState.Companion.OPEN: ReadyState get() = \"open\".asDynamic().unsafeCast<ReadyState>()\n\npublic inline val ReadyState.Companion.ENDED: ReadyState get() = \"ended\".asDynamic().unsafeCast<ReadyState>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface EndOfStreamError {\n   companion object\n}\n\npublic inline val EndOfStreamError.Companion.NETWORK: EndOfStreamError get() = \"network\".asDynamic().unsafeCast<EndOfStreamError>()\n\npublic inline val EndOfStreamError.Companion.DECODE: EndOfStreamError get() = \"decode\".asDynamic().unsafeCast<EndOfStreamError>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface AppendMode {\n   companion object\n}\n\npublic inline val AppendMode.Companion.SEGMENTS: AppendMode get() = \"segments\".asDynamic().unsafeCast<AppendMode>()\n\npublic inline val AppendMode.Companion.SEQUENCE: AppendMode get() = \"sequence\".asDynamic().unsafeCast<AppendMode>()","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.dom.pointerevents\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\nimport org.w3c.dom.events.*\n\npublic external interface PointerEventInit : MouseEventInit {\n   var pointerId: Int? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var width: Double? /* = 1.0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var height: Double? /* = 1.0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var pressure: Float? /* = 0f */\n   get() = definedExternally\n   set(value) = definedExternally\n   var tangentialPressure: Float? /* = 0f */\n   get() = definedExternally\n   set(value) = definedExternally\n   var tiltX: Int? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var tiltY: Int? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var twist: Int? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var pointerType: String? /* = \"\" */\n   get() = definedExternally\n   set(value) = definedExternally\n   var isPrimary: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun PointerEventInit(pointerId: Int? = 0, width: Double? = 1.0, height: Double? = 1.0, pressure: Float? = 0f, tangentialPressure: Float? = 0f, tiltX: Int? = 0, tiltY: Int? = 0, twist: Int? = 0, pointerType: String? = \"\", isPrimary: Boolean? = false, screenX: Int? = 0, screenY: Int? = 0, clientX: Int? = 0, clientY: Int? = 0, button: Short? = 0, buttons: Short? = 0, relatedTarget: EventTarget? = null, region: String? = null, ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false, metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn:

Boolean? = false, modifierFnLock: Boolean? = false, modifierHyper: Boolean? = false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false, modifierSymbol: Boolean? = false, modifierSymbolLock: Boolean? = false, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): PointerEventInit {\n val o = js(\"({})\")\n o[\"pointerId\"] = pointerId\n o[\"width\"] = width\n o[\"height\"] = height\n o[\"pressure\"] = pressure\n o[\"tangentialPressure\"] = tangentialPressure\n o[\"tiltX\"] = tiltX\n o[\"tiltY\"] = tiltY\n o[\"twist\"] = twist\n o[\"pointerType\"] = pointerType\n o[\"isPrimary\"] = isPrimary\n o[\"screenX\"] = screenX\n o[\"screenY\"] = screenY\n o[\"clientX\"] = clientX\n o[\"clientY\"] = clientY\n o[\"button\"] = button\n o[\"buttons\"] = buttons\n o[\"relatedTarget\"] = relatedTarget\n o[\"region\"] = region\n o[\"ctrlKey\"] = ctrlKey\n o[\"shiftKey\"] = shiftKey\n o[\"altKey\"] = altKey\n o[\"metaKey\"] = metaKey\n o[\"modifierAltGraph\"] = modifierAltGraph\n o[\"modifierCapsLock\"] = modifierCapsLock\n o[\"modifierFn\"] = modifierFn\n o[\"modifierFnLock\"] = modifierFnLock\n o[\"modifierHyper\"] = modifierHyper\n o[\"modifierNumLock\"] = modifierNumLock\n o[\"modifierScrollLock\"] = modifierScrollLock\n o[\"modifierSuper\"] = modifierSuper\n o[\"modifierSymbol\"] = modifierSymbol\n o[\"modifierSymbolLock\"] = modifierSymbolLock\n o[\"view\"] = view\n o[\"detail\"] = detail\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [PointerEvent](https://developer.mozilla.org/en/docs/Web/API/PointerEvent) to Kotlin\n */\npublic external open class PointerEvent(type: String, eventInitDict: PointerEventInit = definedExternally) : MouseEvent {\n open val pointerId: Int\n open val width: Double\n open val height: Double\n open val pressure: Float\n open val tangentialPressure: Float\n open val tiltX: Int\n open val tiltY: Int\n open val twist: Int\n open val pointerType: String\n open val isPrimary: Boolean\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.dom.svg\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\nimport org.w3c.dom.css.*\n\n/**\n * Exposes the JavaScript [SVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGElement) to Kotlin\n */\npublic external abstract class SVGElement : Element, ElementCSSInlineStyle, GlobalEventHandlers, SVGElementInstance {\n open val dataset: DOMStringMap\n open val ownerSVGElement: SVGSVGElement?\n open val viewportElement: SVGElement?\n open var tabIndex: Int\n fun focus()\n fun blur()\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external interface SVGBoundingBoxOptions {\n var fill: Boolean? /* = true */\n get() = definedExternally\n set(value) = definedExternally\n var stroke: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n var markers: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n var clipped: Boolean? /* = false */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun SVGBoundingBoxOptions(fill: Boolean? = true, stroke: Boolean? = false, markers: Boolean? = false, clipped: Boolean? = false):

SVGBoundingBoxOptions {\n   val o = js(\"({})\")\n   o[\"fill\"] = fill\n   o[\"stroke\"] = stroke\n   o[\"markers\"] = markers\n   o[\"clipped\"] = clipped\n   return o\n}\n\n/**\n * Exposes the JavaScript [SVGGraphicsElement](https://developer.mozilla.org/en/docs/Web/API/SVGGraphicsElement) to Kotlin\n */\npublic external abstract class SVGGraphicsElement : SVGElement, SVGTests {\n   open val transform: SVGAnimatedTransformList\n   fun getBBox(options: SVGBoundingBoxOptions = definedExternally): DOMRect\n   fun getCTM(): DOMMatrix?\n   fun getScreenCTM(): DOMMatrix?\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [SVGGeometryElement](https://developer.mozilla.org/en/docs/Web/API/SVGGeometryElement) to Kotlin\n */\npublic external abstract class SVGGeometryElement : SVGGraphicsElement {\n   open val pathLength: SVGAnimatedNumber\n   fun isPointInFill(point: DOMPoint): Boolean\n   fun isPointInStroke(point: DOMPoint): Boolean\n   fun getTotalLength(): Float\n   fun getPointAtLength(distance: Float): DOMPoint\n\n   companion object {\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [SVGNumber](https://developer.mozilla.org/en/docs/Web/API/SVGNumber) to Kotlin\n */\npublic external abstract class SVGNumber {\n   open var value: Float\n}\n\n/**\n * Exposes the JavaScript [SVGLength](https://developer.mozilla.org/en/docs/Web/API/SVGLength) to Kotlin\n */\npublic external abstract class SVGLength {\n   open val unitType: Short\n   open var value: Float\n   open var valueInSpecifiedUnits: Float\n   open var valueAsString: String\n   fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)\n   fun convertToSpecifiedUnits(unitType: Short)\n\n   companion object {\n   val SVG_LENGTHTYPE_UNKNOWN: Short\n   val SVG_LENGTHTYPE_NUMBER: Short\n   val SVG_LENGTHTYPE_PERCENTAGE: Short\n   val SVG_LENGTHTYPE_EMS: Short\n   val SVG_LENGTHTYPE_EXS: Short\n   val SVG_LENGTHTYPE_PX: Short\n   val SVG_LENGTHTYPE_CM: Short\n   val SVG_LENGTHTYPE_MM: Short\n   val SVG_LENGTHTYPE_IN: Short\n   val SVG_LENGTHTYPE_PT: Short\n   val SVG_LENGTHTYPE_PC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [SVGAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAngle) to Kotlin\n */\npublic external abstract class SVGAngle {\n   open val unitType: Short\n   open var value: Float\n   open var valueInSpecifiedUnits: Float\n   open var valueAsString: String\n   fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)\n   fun convertToSpecifiedUnits(unitType: Short)\n\n   companion object {\n   val SVG_ANGLETYPE_UNKNOWN: Short\n   val SVG_ANGLETYPE_UNSPECIFIED: Short\n   val SVG_ANGLETYPE_DEG: Short\n   val SVG_ANGLETYPE_RAD: Short\n   val SVG_ANGLETYPE_GRAD: Short\n   }\n}\n\npublic external abstract class SVGNameList {\n   open val length: Int\n   open val numberOfItems: Int\n   fun clear()\n   fun initialize(newItem: dynamic): dynamic\n   fun

insertItemBefore(newItem: dynamic, index: Int): dynamic\n   fun replaceItem(newItem: dynamic, index: Int): dynamic\n   fun removeItem(index: Int): dynamic\n   fun appendItem(newItem: dynamic): dynamic\n   fun getItem(index: Int): dynamic\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGNameList.get(index: Int): dynamic = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGNameList.set(index: Int, newItem: dynamic) { asDynamic()[index] = newItem }\n\n/**\n * Exposes the JavaScript [SVGNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGNumberList) to Kotlin\n */\npublic external abstract class SVGNumberList {\n   open val length: Int\n   open val numberOfItems: Int\n   fun clear()\n   fun initialize(newItem: SVGNumber): SVGNumber\n   fun insertItemBefore(newItem: SVGNumber, index: Int): SVGNumber\n   fun replaceItem(newItem: SVGNumber, index: Int): SVGNumber\n   fun removeItem(index: Int): SVGNumber\n   fun appendItem(newItem: SVGNumber): SVGNumber\n   fun getItem(index: Int): SVGNumber\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGNumberList.get(index: Int): SVGNumber? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGNumberList.set(index: Int, newItem: SVGNumber) { asDynamic()[index] = newItem }\n\n/**\n * Exposes the JavaScript [SVGLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGLengthList) to Kotlin\n */\npublic external abstract class SVGLengthList {\n   open val length: Int\n   open val numberOfItems: Int\n   fun clear()\n   fun initialize(newItem: SVGLength): SVGLength\n   fun insertItemBefore(newItem: SVGLength, index: Int): SVGLength\n   fun replaceItem(newItem: SVGLength, index: Int): SVGLength\n   fun removeItem(index: Int): SVGLength\n   fun appendItem(newItem: SVGLength): SVGLength\n   fun getItem(index: Int): SVGLength\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGLengthList.get(index: Int): SVGLength? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGLengthList.set(index: Int, newItem: SVGLength) { asDynamic()[index] = newItem }\n\n/**\n * Exposes the JavaScript [SVGAnimatedBoolean](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedBoolean) to Kotlin\n */\npublic external abstract class SVGAnimatedBoolean {\n   open var baseVal: Boolean\n   open val animVal: Boolean\n}\n\n/**\n * Exposes the JavaScript [SVGAnimatedEnumeration](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedEnumeration) to Kotlin\n */\npublic external abstract class SVGAnimatedEnumeration {\n   open var baseVal: Short\n   open val animVal: Short\n}\n\n/**\n * Exposes the JavaScript [SVGAnimatedInteger](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedInteger) to Kotlin\n */\npublic external abstract class SVGAnimatedInteger {\n   open var baseVal: Int\n   open val animVal: Int\n}\n\n/**\n * Exposes the JavaScript [SVGAnimatedNumber](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumber) to Kotlin\n */\npublic external abstract class SVGAnimatedNumber {\n   open var baseVal: Float\n   open val animVal: Float\n}\n\n/**\n * Exposes the JavaScript [SVGAnimatedLength](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLength) to Kotlin\n */\npublic external abstract class SVGAnimatedLength {\n   open val baseVal: SVGLength\n   open val animVal: SVGLength\n}\n\n/**\n * Exposes the JavaScript [SVGAnimatedAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedAngle) to Kotlin\n */\npublic external abstract class SVGAnimatedAngle {\n   open val baseVal: SVGAngle\n   open val animVal: SVGAngle\n}\n\n/**\n * Exposes the JavaScript [SVGAnimatedString](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedString) to Kotlin\n */\npublic external abstract class SVGAnimatedString {\n   open var baseVal: String\n   open val animVal: String\n}\n\n/**\n * Exposes the JavaScript [SVGAnimatedRect](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedRect)

to Kotlin\n */\npublic external abstract class SVGAnimatedRect {\n   open val baseVal: DOMRect\n   open val animVal: DOMRectReadOnly\n}\n\n/**\n * Exposes the JavaScript [SVGAnimatedNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumberList) to Kotlin\n */\npublic external abstract class SVGAnimatedNumberList {\n   open val baseVal: SVGNumberList\n   open val animVal: SVGNumberList\n}\n\n/**\n * Exposes the JavaScript [SVGAnimatedLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLengthList) to Kotlin\n */\npublic external abstract class SVGAnimatedLengthList {\n   open val baseVal: SVGLengthList\n   open val animVal: SVGLengthList\n}\n\n/**\n * Exposes the JavaScript [SVGStringList](https://developer.mozilla.org/en/docs/Web/API/SVGStringList) to Kotlin\n */\npublic external abstract class SVGStringList {\n   open val length: Int\n   open val numberOfItems: Int\n   fun clear()\n   fun initialize(newItem: String): String\n   fun insertItemBefore(newItem: String, index: Int): String\n   fun replaceItem(newItem: String, index: Int): String\n   fun removeItem(index: Int): String\n   fun appendItem(newItem: String): String\n   fun getItem(index: Int): String\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGStringList.get(index: Int): String? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGStringList.set(index: Int, newItem: String) { asDynamic()[index] = newItem }\n\n/**\n * Exposes the JavaScript [SVGUnitTypes](https://developer.mozilla.org/en/docs/Web/API/SVGUnitTypes) to Kotlin\n */\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface SVGUnitTypes {\n   companion object {\n   val SVG_UNIT_TYPE_UNKNOWN: Short\n   val SVG_UNIT_TYPE_USERSPACEONUSE: Short\n   val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [SVGTests](https://developer.mozilla.org/en/docs/Web/API/SVGTests) to Kotlin\n */\npublic external interface SVGTests {\n   val requiredExtensions: SVGStringList\n   val systemLanguage: SVGStringList\n}\n\npublic external interface SVGFitToViewBox {\n   val viewBox: SVGAnimatedRect\n   val preserveAspectRatio: SVGAnimatedPreserveAspectRatio\n}\n\n/**\n * Exposes the JavaScript [SVGZoomAndPan](https://developer.mozilla.org/en/docs/Web/API/SVGZoomAndPan) to Kotlin\n */\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface SVGZoomAndPan {\n   var zoomAndPan: Short\n\n   companion object {\n   val SVG_ZOOMANDPAN_UNKNOWN: Short\n   val SVG_ZOOMANDPAN_DISABLE: Short\n   val SVG_ZOOMANDPAN_MAGNIFY: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [SVGURIReference](https://developer.mozilla.org/en/docs/Web/API/SVGURIReference) to Kotlin\n */\npublic external interface SVGURIReference {\n   val href: SVGAnimatedString\n}\n\n/**\n * Exposes the JavaScript [SVGSVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGSVGElement) to Kotlin\n */\npublic external abstract class SVGSVGElement : SVGGraphicsElement, SVGFitToViewBox, SVGZoomAndPan, WindowEventHandlers {\n   open val x: SVGAnimatedLength\n   open val y: SVGAnimatedLength\n   open val width: SVGAnimatedLength\n   open val height: SVGAnimatedLength\n   open var currentScale: Float\n   open val currentTranslate: DOMPointReadOnly\n   fun getIntersectionList(rect: DOMRectReadOnly, referenceElement: SVGElement?): NodeList\n   fun getEnclosureList(rect: DOMRectReadOnly, referenceElement: SVGElement?): NodeList\n   fun checkIntersection(element: SVGElement, rect: DOMRectReadOnly): Boolean\n   fun checkEnclosure(element: SVGElement, rect: DOMRectReadOnly): Boolean\n   fun deselectAll()\n   fun createSVGNumber(): SVGNumber\n   fun createSVGLength(): SVGLength\n   fun createSVGAngle(): SVGAngle\n   fun createSVGPoint(): DOMPoint\n   fun createSVGMatrix(): DOMMatrix\n   fun createSVGRect(): DOMRect\n   fun createSVGTransform(): SVGTransform\n   fun createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly): SVGTransform\n   fun getElementById(elementId: String): Element\n   fun suspendRedraw(maxWaitMilliseconds: Int): Int\n   fun unsuspendRedraw(suspendHandleID: Int)\n   fun unsuspendRedrawAll()\n   fun forceRedraw()\n\n   companion

object {\n val SVG_ZOOMANDPAN_UNKNOWN: Short\n val SVG_ZOOMANDPAN_DISABLE: Short\n val SVG_ZOOMANDPAN_MAGNIFY: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGGElement](https://developer.mozilla.org/en/docs/Web/API/SVGGElement) to Kotlin\n */\npublic external abstract class SVGGElement : SVGGraphicsElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external abstract class SVGUnknownElement : SVGGraphicsElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGDefsElement](https://developer.mozilla.org/en/docs/Web/API/SVGDefsElement) to Kotlin\n */\npublic external abstract class SVGDefsElement : SVGGraphicsElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGDescElement](https://developer.mozilla.org/en/docs/Web/API/SVGDescElement) to Kotlin\n */\npublic external abstract class SVGDescElement : SVGElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val

DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGMetadataElement](https://developer.mozilla.org/en/docs/Web/API/SVGMetadataElement) to Kotlin\n */\npublic external abstract class SVGMetadataElement : SVGElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGTitleElement](https://developer.mozilla.org/en/docs/Web/API/SVGTitleElement) to Kotlin\n */\npublic external abstract class SVGTitleElement : SVGElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGSymbolElement](https://developer.mozilla.org/en/docs/Web/API/SVGSymbolElement) to Kotlin\n */\npublic external abstract class SVGSymbolElement : SVGGraphicsElement, SVGFitToViewBox {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGUseElement](https://developer.mozilla.org/en/docs/Web/API/SVGUseElement) to Kotlin\n */\npublic external abstract class SVGUseElement : SVGGraphicsElement, SVGURIReference {\n open val x: SVGAnimatedLength\n open val y: SVGAnimatedLength\n open val width: SVGAnimatedLength\n open val height: SVGAnimatedLength\n open val instanceRoot: SVGElement?\n open val animatedInstanceRoot: SVGElement?\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val

DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external open class SVGUseElementShadowRoot : ShadowRoot {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external interface SVGElementInstance {\n val correspondingElement: SVGElement?\n get() = definedExternally\n val correspondingUseElement: SVGUseElement?\n get() = definedExternally\n}\n\npublic external open class ShadowAnimation(source: dynamic, newTarget: dynamic) {\n open val sourceAnimation: dynamic\n}\n\n/**\n * Exposes the JavaScript [SVGSwitchElement](https://developer.mozilla.org/en/docs/Web/API/SVGSwitchElement) to Kotlin\n */\npublic external abstract class SVGSwitchElement : SVGGraphicsElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external interface GetSVGDocument {\n fun getSVGDocument(): Document\n}\n\n/**\n * Exposes the JavaScript [SVGStyleElement](https://developer.mozilla.org/en/docs/Web/API/SVGStyleElement) to Kotlin\n */\npublic external abstract class SVGStyleElement : SVGElement, LinkStyle {\n open var type: String\n open var media: String\n open var title: String\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGTransform](https://developer.mozilla.org/en/docs/Web/API/SVGTransform) to Kotlin\n */\npublic external abstract class SVGTransform {\n open val type: Short\n open val matrix: DOMMatrix\n open val angle: Float\n fun setMatrix(matrix: DOMMatrixReadOnly)\n fun setTranslate(tx: Float, ty: Float)\n fun setScale(sx: Float, sy: Float)\n fun setRotate(angle: Float, cx: Float, cy: Float)\n fun setSkewX(angle: Float)\n fun setSkewY(angle: Float)\n\n companion object {\n val SVG_TRANSFORM_UNKNOWN: Short\n val SVG_TRANSFORM_MATRIX: Short\n val SVG_TRANSFORM_TRANSLATE: Short\n val

SVG_TRANSFORM_SCALE: Short\n val SVG_TRANSFORM_ROTATE: Short\n val SVG_TRANSFORM_SKEWX: Short\n val SVG_TRANSFORM_SKEWY: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin\n */\npublic external abstract class SVGTransformList {\n open val length: Int\n open val numberOfItems: Int\n fun clear()\n fun initialize(newItem: SVGTransform): SVGTransform\n fun insertItemBefore(newItem: SVGTransform, index: Int): SVGTransform\n fun replaceItem(newItem: SVGTransform, index: Int): SVGTransform\n fun removeItem(index: Int): SVGTransform\n fun appendItem(newItem: SVGTransform): SVGTransform\n fun createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly): SVGTransform\n fun consolidate(): SVGTransform?\n fun getItem(index: Int): SVGTransform\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGTransformList.get(index: Int): SVGTransform? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGTransformList.set(index: Int, newItem: SVGTransform) { asDynamic()[index] = newItem }\n\n/**\n * Exposes the JavaScript [SVGAnimatedTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedTransformList) to Kotlin\n */\npublic external abstract class SVGAnimatedTransformList {\n open val baseVal: SVGTransformList\n open val animVal: SVGTransformList\n}\n\n/**\n * Exposes the JavaScript [SVGPreserveAspectRatio](https://developer.mozilla.org/en/docs/Web/API/SVGPreserveAspectRatio) to Kotlin\n */\npublic external abstract class SVGPreserveAspectRatio {\n open var align: Short\n open var meetOrSlice: Short\n\n companion object {\n val SVG_PRESERVEASPECTRATIO_UNKNOWN: Short\n val SVG_PRESERVEASPECTRATIO_NONE: Short\n val SVG_PRESERVEASPECTRATIO_XMINYMIN: Short\n val SVG_PRESERVEASPECTRATIO_XMIDYMIN: Short\n val SVG_PRESERVEASPECTRATIO_XMAXYMIN: Short\n val SVG_PRESERVEASPECTRATIO_XMINYMID: Short\n val SVG_PRESERVEASPECTRATIO_XMIDYMID: Short\n val SVG_PRESERVEASPECTRATIO_XMAXYMID: Short\n val SVG_PRESERVEASPECTRATIO_XMINYMAX: Short\n val SVG_PRESERVEASPECTRATIO_XMIDYMAX: Short\n val SVG_PRESERVEASPECTRATIO_XMAXYMAX: Short\n val SVG_MEETORSLICE_UNKNOWN: Short\n val SVG_MEETORSLICE_MEET: Short\n val SVG_MEETORSLICE_SLICE: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGAnimatedPreserveAspectRatio](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPreserveAspectRatio) to Kotlin\n */\npublic external abstract class SVGAnimatedPreserveAspectRatio {\n open val baseVal: SVGPreserveAspectRatio\n open val animVal: SVGPreserveAspectRatio\n}\n\n/**\n * Exposes the JavaScript [SVGPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGPathElement) to Kotlin\n */\npublic external abstract class SVGPathElement : SVGGeometryElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGRectElement](https://developer.mozilla.org/en/docs/Web/API/SVGRectElement) to Kotlin\n */\npublic external abstract class SVGRectElement : SVGGeometryElement {\n open val x: SVGAnimatedLength\n open val y: SVGAnimatedLength\n open val width: SVGAnimatedLength\n open val height: SVGAnimatedLength\n

```
 open val rx: SVGAnimatedLength\n    open val ry: SVGAnimatedLength\n\n    companion object {\n        val
ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val
CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE:
Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[SVGCircleElement](https://developer.mozilla.org/en/docs/Web/API/SVGCircleElement) to Kotlin\n */\npublic
external abstract class SVGCircleElement : SVGGeometryElement {\n    open val cx: SVGAnimatedLength\n
open val cy: SVGAnimatedLength\n    open val r: SVGAnimatedLength\n\n    companion object {\n        val
ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val
CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE:
Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[SVGEllipseElement](https://developer.mozilla.org/en/docs/Web/API/SVGEllipseElement) to Kotlin\n */\npublic
external abstract class SVGEllipseElement : SVGGeometryElement {\n    open val cx: SVGAnimatedLength\n
open val cy: SVGAnimatedLength\n    open val rx: SVGAnimatedLength\n    open val ry: SVGAnimatedLength\n\n
    companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val
TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE:
Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val
COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n
    val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[SVGLineElement](https://developer.mozilla.org/en/docs/Web/API/SVGLineElement) to Kotlin\n */\npublic
external abstract class SVGLineElement : SVGGeometryElement {\n    open val x1: SVGAnimatedLength\n    open
val y1: SVGAnimatedLength\n    open val x2: SVGAnimatedLength\n    open val y2: SVGAnimatedLength\n\n
companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val
TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE:
Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val
COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n
    val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[SVGMeshElement](https://developer.mozilla.org/en/docs/Web/API/SVGMeshElement) to Kotlin\n */\npublic
external abstract class SVGMeshElement : SVGGeometryElement, SVGURIReference {\n    companion object {\n
```

```
    val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val
CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE:
Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[SVGAnimatedPoints](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPoints) to Kotlin\n */\npublic
external interface SVGAnimatedPoints {\n    val points: SVGPointList\n    val animatedPoints:
SVGPointList\n}\n\npublic external abstract class SVGPointList {\n    open val length: Int\n    open val
numberOfItems: Int\n    fun clear()\n    fun initialize(newItem: DOMPoint): DOMPoint\n    fun
insertItemBefore(newItem: DOMPoint, index: Int): DOMPoint\n    fun replaceItem(newItem: DOMPoint, index:
Int): DOMPoint\n    fun removeItem(index: Int): DOMPoint\n    fun appendItem(newItem: DOMPoint):
DOMPoint\n    fun getItem(index: Int): DOMPoint\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGPointList.get(index: Int):
DOMPoint? = asDynamic()[index]\n\n@Suppress(\"INVISIBLE_REFERENCE\",
\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun SVGPointList.set(index: Int,
newItem: DOMPoint) { asDynamic()[index] = newItem }\n\n/**\n * Exposes the JavaScript
[SVGPolylineElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolylineElement) to Kotlin\n
*/\npublic external abstract class SVGPolylineElement : SVGGeometryElement, SVGAnimatedPoints {\n
companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val
TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE:
Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val
COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n
    val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[SVGPolygonElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolygonElement) to Kotlin\n
*/\npublic external abstract class SVGPolygonElement : SVGGeometryElement, SVGAnimatedPoints {\n
companion object {\n        val ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val
TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE:
Short\n        val ENTITY_NODE: Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val
COMMENT_NODE: Short\n        val DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n
    val DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[SVGTextContentElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextContentElement) to Kotlin\n
*/\npublic external abstract class SVGTextContentElement : SVGGraphicsElement {\n    open val textLength:
SVGAnimatedLength\n    open val lengthAdjust: SVGAnimatedEnumeration\n    fun getNumberOfChars(): Int\n
fun getComputedTextLength(): Float\n    fun getSubStringLength(charnum: Int, nchars: Int): Float\n    fun
getStartPositionOfChar(charnum: Int): DOMPoint\n    fun getEndPositionOfChar(charnum: Int): DOMPoint\n    fun
getExtentOfChar(charnum: Int): DOMRect\n    fun getRotationOfChar(charnum: Int): Float\n    fun
```

getCharNumAtPosition(point: DOMPoint): Int\n   fun selectSubString(charnum: Int, nchars: Int)\n\n   companion object {\n   val LENGTHADJUST_UNKNOWN: Short\n   val LENGTHADJUST_SPACING: Short\n   val LENGTHADJUST_SPACINGANDGLYPHS: Short\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [SVGTextPositioningElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextPositioningElement) to Kotlin\n */\npublic external abstract class SVGTextPositioningElement : SVGTextContentElement {\n   open val x: SVGAnimatedLengthList\n   open val y: SVGAnimatedLengthList\n   open val dx: SVGAnimatedLengthList\n   open val dy: SVGAnimatedLengthList\n   open val rotate: SVGAnimatedNumberList\n\n   companion object {\n   val LENGTHADJUST_UNKNOWN: Short\n   val LENGTHADJUST_SPACING: Short\n   val LENGTHADJUST_SPACINGANDGLYPHS: Short\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [SVGTextElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextElement) to Kotlin\n */\npublic external abstract class SVGTextElement : SVGTextPositioningElement {\n   companion object {\n   val LENGTHADJUST_UNKNOWN: Short\n   val LENGTHADJUST_SPACING: Short\n   val LENGTHADJUST_SPACINGANDGLYPHS: Short\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val DOCUMENT_FRAGMENT_NODE: Short\n   val NOTATION_NODE: Short\n   val DOCUMENT_POSITION_DISCONNECTED: Short\n   val DOCUMENT_POSITION_PRECEDING: Short\n   val DOCUMENT_POSITION_FOLLOWING: Short\n   val DOCUMENT_POSITION_CONTAINS: Short\n   val DOCUMENT_POSITION_CONTAINED_BY: Short\n   val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [SVGTSpanElement](https://developer.mozilla.org/en/docs/Web/API/SVGTSpanElement) to Kotlin\n */\npublic external abstract class SVGTSpanElement : SVGTextPositioningElement {\n   companion object {\n   val LENGTHADJUST_UNKNOWN: Short\n   val LENGTHADJUST_SPACING: Short\n   val LENGTHADJUST_SPACINGANDGLYPHS: Short\n   val ELEMENT_NODE: Short\n   val ATTRIBUTE_NODE: Short\n   val TEXT_NODE: Short\n   val CDATA_SECTION_NODE: Short\n   val ENTITY_REFERENCE_NODE: Short\n   val ENTITY_NODE: Short\n   val PROCESSING_INSTRUCTION_NODE: Short\n   val COMMENT_NODE: Short\n   val DOCUMENT_NODE: Short\n   val DOCUMENT_TYPE_NODE: Short\n   val

DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGTextPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextPathElement) to Kotlin\n */\npublic external abstract class SVGTextPathElement : SVGTextContentElement, SVGURIReference {\n open val startOffset: SVGAnimatedLength\n open val method: SVGAnimatedEnumeration\n open val spacing: SVGAnimatedEnumeration\n\n companion object {\n val TEXTPATH_METHODTYPE_UNKNOWN: Short\n val TEXTPATH_METHODTYPE_ALIGN: Short\n val TEXTPATH_METHODTYPE_STRETCH: Short\n val TEXTPATH_SPACINGTYPE_UNKNOWN: Short\n val TEXTPATH_SPACINGTYPE_AUTO: Short\n val TEXTPATH_SPACINGTYPE_EXACT: Short\n val LENGTHADJUST_UNKNOWN: Short\n val LENGTHADJUST_SPACING: Short\n val LENGTHADJUST_SPACINGANDGLYPHS: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGImageElement](https://developer.mozilla.org/en/docs/Web/API/SVGImageElement) to Kotlin\n */\npublic external abstract class SVGImageElement : SVGGraphicsElement, SVGURIReference, HTMLOrSVGImageElement {\n open val x: SVGAnimatedLength\n open val y: SVGAnimatedLength\n open val width: SVGAnimatedLength\n open val height: SVGAnimatedLength\n open val preserveAspectRatio: SVGAnimatedPreserveAspectRatio\n open var crossOrigin: String?\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGForeignObjectElement](https://developer.mozilla.org/en/docs/Web/API/SVGForeignObjectElement) to Kotlin\n */\npublic external abstract class SVGForeignObjectElement : SVGGraphicsElement {\n open val x: SVGAnimatedLength\n open val y: SVGAnimatedLength\n open val width: SVGAnimatedLength\n open val height: SVGAnimatedLength\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n

```
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\npublic external abstract class
SVGMarkerElement : SVGElement, SVGFitToViewBox {\n    open val refX: SVGAnimatedLength\n    open val
refY: SVGAnimatedLength\n    open val markerUnits: SVGAnimatedEnumeration\n    open val markerWidth:
SVGAnimatedLength\n    open val markerHeight: SVGAnimatedLength\n    open val orientType:
SVGAnimatedEnumeration\n    open val orientAngle: SVGAnimatedAngle\n    open var orient: String\n    fun
setOrientToAuto()\n    fun setOrientToAngle(angle: SVGAngle)\n\n    companion object {\n        val
SVG_MARKERUNITS_UNKNOWN: Short\n        val SVG_MARKERUNITS_USERSPACEONUSE: Short\n
val SVG_MARKERUNITS_STROKEWIDTH: Short\n        val SVG_MARKER_ORIENT_UNKNOWN: Short\n
    val SVG_MARKER_ORIENT_AUTO: Short\n        val SVG_MARKER_ORIENT_ANGLE: Short\n        val
ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val
CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE:
Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[SVGSolidcolorElement](https://developer.mozilla.org/en/docs/Web/API/SVGSolidcolorElement) to Kotlin\n
*/\npublic external abstract class SVGSolidcolorElement : SVGElement {\n    companion object {\n        val
ELEMENT_NODE: Short\n        val ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val
CDATA_SECTION_NODE: Short\n        val ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE:
Short\n        val PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[SVGGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGGradientElement) to Kotlin\n
*/\npublic external abstract class SVGGradientElement : SVGElement, SVGURIReference, SVGUnitTypes {\n
open val gradientUnits: SVGAnimatedEnumeration\n    open val gradientTransform: SVGAnimatedTransformList\n
    open val spreadMethod: SVGAnimatedEnumeration\n\n    companion object {\n        val
SVG_SPREADMETHOD_UNKNOWN: Short\n        val SVG_SPREADMETHOD_PAD: Short\n        val
SVG_SPREADMETHOD_REFLECT: Short\n        val SVG_SPREADMETHOD_REPEAT: Short\n        val
SVG_UNIT_TYPE_UNKNOWN: Short\n        val SVG_UNIT_TYPE_USERSPACEONUSE: Short\n        val
SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short\n        val ELEMENT_NODE: Short\n        val
ATTRIBUTE_NODE: Short\n        val TEXT_NODE: Short\n        val CDATA_SECTION_NODE: Short\n        val
ENTITY_REFERENCE_NODE: Short\n        val ENTITY_NODE: Short\n        val
PROCESSING_INSTRUCTION_NODE: Short\n        val COMMENT_NODE: Short\n        val
DOCUMENT_NODE: Short\n        val DOCUMENT_TYPE_NODE: Short\n        val
DOCUMENT_FRAGMENT_NODE: Short\n        val NOTATION_NODE: Short\n        val
DOCUMENT_POSITION_DISCONNECTED: Short\n        val DOCUMENT_POSITION_PRECEDING: Short\n
    val DOCUMENT_POSITION_FOLLOWING: Short\n        val DOCUMENT_POSITION_CONTAINS: Short\n
    val DOCUMENT_POSITION_CONTAINED_BY: Short\n        val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n    }\n}\n\n/**\n * Exposes the JavaScript
[SVGLinearGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGLinearGradientElement) to
```

Kotlin\n */\npublic external abstract class SVGLinearGradientElement : SVGGradientElement {\n open val x1: SVGAnimatedLength\n open val y1: SVGAnimatedLength\n open val x2: SVGAnimatedLength\n open val y2: SVGAnimatedLength\n\n companion object {\n val SVG_SPREADMETHOD_UNKNOWN: Short\n val SVG_SPREADMETHOD_PAD: Short\n val SVG_SPREADMETHOD_REFLECT: Short\n val SVG_SPREADMETHOD_REPEAT: Short\n val SVG_UNIT_TYPE_UNKNOWN: Short\n val SVG_UNIT_TYPE_USERSPACEONUSE: Short\n val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGRadialGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGRadialGradientElement) to Kotlin\n */\npublic external abstract class SVGRadialGradientElement : SVGGradientElement {\n open val cx: SVGAnimatedLength\n open val cy: SVGAnimatedLength\n open val r: SVGAnimatedLength\n open val fx: SVGAnimatedLength\n open val fy: SVGAnimatedLength\n open val fr: SVGAnimatedLength\n\n companion object {\n val SVG_SPREADMETHOD_UNKNOWN: Short\n val SVG_SPREADMETHOD_PAD: Short\n val SVG_SPREADMETHOD_REFLECT: Short\n val SVG_SPREADMETHOD_REPEAT: Short\n val SVG_UNIT_TYPE_UNKNOWN: Short\n val SVG_UNIT_TYPE_USERSPACEONUSE: Short\n val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external abstract class SVGMeshGradientElement : SVGGradientElement {\n companion object {\n val SVG_SPREADMETHOD_UNKNOWN: Short\n val SVG_SPREADMETHOD_PAD: Short\n val SVG_SPREADMETHOD_REFLECT: Short\n val SVG_SPREADMETHOD_REPEAT: Short\n val SVG_UNIT_TYPE_UNKNOWN: Short\n val SVG_UNIT_TYPE_USERSPACEONUSE: Short\n val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external abstract class SVGMeshrowElement : SVGElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val

ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external abstract class SVGMeshpatchElement : SVGElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGStopElement](https://developer.mozilla.org/en/docs/Web/API/SVGStopElement) to Kotlin\n */\npublic external abstract class SVGStopElement : SVGElement {\n open val offset: SVGAnimatedNumber\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGPatternElement](https://developer.mozilla.org/en/docs/Web/API/SVGPatternElement) to Kotlin\n */\npublic external abstract class SVGPatternElement : SVGElement, SVGFitToViewBox, SVGURIReference, SVGUnitTypes {\n open val patternUnits: SVGAnimatedEnumeration\n open val patternContentUnits: SVGAnimatedEnumeration\n open val patternTransform: SVGAnimatedTransformList\n open val x: SVGAnimatedLength\n open val y: SVGAnimatedLength\n open val width: SVGAnimatedLength\n open val height: SVGAnimatedLength\n\n companion object {\n val SVG_UNIT_TYPE_UNKNOWN: Short\n val SVG_UNIT_TYPE_USERSPACEONUSE: Short\n val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external abstract class SVGHatchElement : SVGElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val

PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external abstract class SVGHatchpathElement : SVGElement {\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGCursorElement](https://developer.mozilla.org/en/docs/Web/API/SVGCursorElement) to Kotlin\n */\npublic external abstract class SVGCursorElement : SVGElement, SVGURIReference {\n open val x: SVGAnimatedLength\n open val y: SVGAnimatedLength\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGScriptElement](https://developer.mozilla.org/en/docs/Web/API/SVGScriptElement) to Kotlin\n */\npublic external abstract class SVGScriptElement : SVGElement, SVGURIReference, HTMLOrSVGScriptElement {\n open var type: String\n open var crossOrigin: String?\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGAElement](https://developer.mozilla.org/en/docs/Web/API/SVGAElement) to Kotlin\n */\npublic external abstract class SVGAElement : SVGGraphicsElement, SVGURIReference {\n open val target: SVGAnimatedString\n open val download: SVGAnimatedString\n open val rel: SVGAnimatedString\n open val relList: SVGAnimatedString\n open val hreflang: SVGAnimatedString\n open val type: SVGAnimatedString\n\n companion object {\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val

PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\n/**\n * Exposes the JavaScript [SVGViewElement](https://developer.mozilla.org/en/docs/Web/API/SVGViewElement) to Kotlin\n */\npublic external abstract class SVGViewElement : SVGElement, SVGFitToViewBox, SVGZoomAndPan {\n companion object {\n val SVG_ZOOMANDPAN_UNKNOWN: Short\n val SVG_ZOOMANDPAN_DISABLE: Short\n val SVG_ZOOMANDPAN_MAGNIFY: Short\n val ELEMENT_NODE: Short\n val ATTRIBUTE_NODE: Short\n val TEXT_NODE: Short\n val CDATA_SECTION_NODE: Short\n val ENTITY_REFERENCE_NODE: Short\n val ENTITY_NODE: Short\n val PROCESSING_INSTRUCTION_NODE: Short\n val COMMENT_NODE: Short\n val DOCUMENT_NODE: Short\n val DOCUMENT_TYPE_NODE: Short\n val DOCUMENT_FRAGMENT_NODE: Short\n val NOTATION_NODE: Short\n val DOCUMENT_POSITION_DISCONNECTED: Short\n val DOCUMENT_POSITION_PRECEDING: Short\n val DOCUMENT_POSITION_FOLLOWING: Short\n val DOCUMENT_POSITION_CONTAINS: Short\n val DOCUMENT_POSITION_CONTAINED_BY: Short\n val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.files\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\nimport org.w3c.dom.events.*\nimport org.w3c.xhr.*\n\n/**\n * Exposes the JavaScript [Blob](https://developer.mozilla.org/en/docs/Web/API/Blob) to Kotlin\n */\npublic external open class Blob(blobParts: Array<dynamic> = definedExternally, options: BlobPropertyBag = definedExternally) : MediaProvider, ImageBitmapSource {\n open val size: Number\n open val type: String\n open val isClosed: Boolean\n fun slice(start: Int = definedExternally, end: Int = definedExternally, contentType: String = definedExternally): Blob\n fun close()\n}\n\npublic external interface BlobPropertyBag {\n var type: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun BlobPropertyBag(type: String? = \"\"): BlobPropertyBag {\n val o = js(\"({})\")\n o[\"type\"] = type\n return o\n}\n\n/**\n * Exposes the JavaScript [File](https://developer.mozilla.org/en/docs/Web/API/File) to Kotlin\n */\npublic external open class File(fileBits: Array<dynamic>, fileName: String, options: FilePropertyBag = definedExternally) : Blob {\n open val name: String\n open val lastModified: Int\n}\n\npublic external interface FilePropertyBag : BlobPropertyBag {\n var lastModified: Int?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun FilePropertyBag(lastModified: Int? = undefined, type: String? = \"\"): FilePropertyBag {\n val o = js(\"({})\")\n o[\"lastModified\"] = lastModified\n o[\"type\"] = type\n return o\n}\n\n/**\n * Exposes the JavaScript [FileList](https://developer.mozilla.org/en/docs/Web/API/FileList) to Kotlin\n */\npublic external abstract class FileList : ItemArrayLike<File> {\n override fun item(index: Int): File?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline operator fun FileList.get(index: Int): File? = asDynamic()[index]\n\n/**\n * Exposes the JavaScript [FileReader](https://developer.mozilla.org/en/docs/Web/API/FileReader) to Kotlin\n */\npublic external open class FileReader : EventTarget {\n open val readyState: Short\n open val result: dynamic\n open val error:

dynamic\n   var onloadstart: ((ProgressEvent) -> dynamic)?\n   var onprogress: ((ProgressEvent) -> dynamic)?\n   var onload: ((Event) -> dynamic)?\n   var onabort: ((Event) -> dynamic)?\n   var onerror: ((Event) -> dynamic)?\n   var onloadend: ((Event) -> dynamic)?\n   fun readAsArrayBuffer(blob: Blob)\n   fun readAsBinaryString(blob: Blob)\n   fun readAsText(blob: Blob, label: String = definedExternally)\n   fun readAsDataURL(blob: Blob)\n   fun abort()\n\n   companion object {\n   val EMPTY: Short\n   val LOADING: Short\n   val DONE: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [FileReaderSync](https://developer.mozilla.org/en/docs/Web/API/FileReaderSync) to Kotlin\n */\npublic external open class FileReaderSync {\n   fun readAsArrayBuffer(blob: Blob): ArrayBuffer\n   fun readAsBinaryString(blob: Blob): String\n   fun readAsText(blob: Blob, label: String = definedExternally): String\n   fun readAsDataURL(blob: Blob): String\n}","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.notifications\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.events.*\nimport org.w3c.workers.*\n\n/**\n * Exposes the JavaScript [Notification](https://developer.mozilla.org/en/docs/Web/API/Notification) to Kotlin\n */\npublic external open class Notification(title: String, options: NotificationOptions = definedExternally) : EventTarget {\n   var onclick: ((MouseEvent) -> dynamic)?\n   var onerror: ((Event) -> dynamic)?\n   open val title: String\n   open val dir: NotificationDirection\n   open val lang: String\n   open val body: String\n   open val tag: String\n   open val image: String\n   open val icon: String\n   open val badge: String\n   open val sound: String\n   open val vibrate: Array<out Int>\n   open val timestamp: Number\n   open val renotify: Boolean\n   open val silent: Boolean\n   open val noscreen: Boolean\n   open val requireInteraction: Boolean\n   open val sticky: Boolean\n   open val data: Any?\n   open val actions: Array<out NotificationAction>\n   fun close()\n\n   companion object {\n   val permission: NotificationPermission\n   val maxActions: Int\n   fun requestPermission(deprecatedCallback: (NotificationPermission) -> Unit = definedExternally): Promise<NotificationPermission>\n   }\n}\n\npublic external interface NotificationOptions {\n   var dir: NotificationDirection? /* = NotificationDirection.AUTO */\n   get() = definedExternally\n   set(value) = definedExternally\n   var lang: String? /* = \"\" */\n   get() = definedExternally\n   set(value) = definedExternally\n   var body: String? /* = \"\" */\n   get() = definedExternally\n   set(value) = definedExternally\n   var tag: String? /* = \"\" */\n   get() = definedExternally\n   set(value) = definedExternally\n   var image: String?\n   get() = definedExternally\n   set(value) = definedExternally\n   var icon: String?\n   get() = definedExternally\n   set(value) = definedExternally\n   var badge: String?\n   get() = definedExternally\n   set(value) = definedExternally\n   var sound: String?\n   get() = definedExternally\n   set(value) = definedExternally\n   var vibrate: dynamic\n   get() = definedExternally\n   set(value) = definedExternally\n   var timestamp: Number?\n   get() = definedExternally\n   set(value) = definedExternally\n   var renotify: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var silent: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var noscreen: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var requireInteraction: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var sticky: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var data: Any? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n   var actions: Array<NotificationAction>? /* = arrayOf() */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun NotificationOptions(dir: NotificationDirection? = NotificationDirection.AUTO, lang: String? = \"\", body: String? = \"\", tag: String? = \"\", image: String? = undefined, icon: String? = undefined, badge: String? = undefined, sound: String? = undefined, vibrate: dynamic = undefined, timestamp: Number? = undefined, renotify: Boolean? = false, silent: Boolean? = false, noscreen: Boolean? = false, requireInteraction: Boolean? = false, sticky: Boolean? = false, data: Any? = null, actions: Array<NotificationAction>? = arrayOf()): NotificationOptions {\n   val o = js(\"({})\")\n   o[\"dir\"] = dir\n

o[\"lang\"] = lang\n o[\"body\"] = body\n o[\"tag\"] = tag\n o[\"image\"] = image\n o[\"icon\"] = icon\n o[\"badge\"] = badge\n o[\"sound\"] = sound\n o[\"vibrate\"] = vibrate\n o[\"timestamp\"] = timestamp\n o[\"renotify\"] = renotify\n o[\"silent\"] = silent\n o[\"noscreen\"] = noscreen\n o[\"requireInteraction\"] = requireInteraction\n o[\"sticky\"] = sticky\n o[\"data\"] = data\n o[\"actions\"] = actions\n return o\n}\n\npublic external interface NotificationAction {\n var action: String?\n var title: String?\n var icon: String?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun NotificationAction(action: String?, title: String?, icon: String? = undefined): NotificationAction {\n val o = js(\"({})\")\n o[\"action\"] = action\n o[\"title\"] = title\n o[\"icon\"] = icon\n return o\n}\n\npublic external interface GetNotificationOptions {\n var tag: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun GetNotificationOptions(tag: String? = \"\"): GetNotificationOptions {\n val o = js(\"({})\")\n o[\"tag\"] = tag\n return o\n}\n\n/**\n * Exposes the JavaScript [NotificationEvent](https://developer.mozilla.org/en/docs/Web/API/NotificationEvent) to Kotlin\n */\npublic external open class NotificationEvent(type: String, eventInitDict: NotificationEventInit) : ExtendableEvent {\n open val notification: Notification\n open val action: String\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface NotificationEventInit : ExtendableEventInit {\n var notification: Notification?\n var action: String? /* = \"\" */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun NotificationEventInit(notification: Notification?, action: String? = \"\", bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): NotificationEventInit {\n val o = js(\"({})\")\n o[\"notification\"] = notification\n o[\"action\"] = action\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface NotificationPermission {\n companion object\n}\n\npublic inline val NotificationPermission.Companion.DEFAULT: NotificationPermission get() = \"default\".asDynamic().unsafeCast<NotificationPermission>()\n\npublic inline val NotificationPermission.Companion.DENIED: NotificationPermission get() = \"denied\".asDynamic().unsafeCast<NotificationPermission>()\n\npublic inline val NotificationPermission.Companion.GRANTED: NotificationPermission get() = \"granted\".asDynamic().unsafeCast<NotificationPermission>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface NotificationDirection {\n companion object\n}\n\npublic inline val NotificationDirection.Companion.AUTO: NotificationDirection get() = \"auto\".asDynamic().unsafeCast<NotificationDirection>()\n\npublic inline val NotificationDirection.Companion.LTR: NotificationDirection get() = \"ltr\".asDynamic().unsafeCast<NotificationDirection>()\n\npublic inline val NotificationDirection.Companion.RTL: NotificationDirection get() = \"rtl\".asDynamic().unsafeCast<NotificationDirection>()","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.workers\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\nimport org.w3c.dom.events.*\nimport org.w3c.fetch.*\nimport org.w3c.notifications.*\n\n/**\n * Exposes the JavaScript [ServiceWorker](https://developer.mozilla.org/en/docs/Web/API/ServiceWorker) to Kotlin\n */\npublic external

abstract class ServiceWorker : EventTarget, AbstractWorker, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {\n   open val scriptURL: String\n   open val state: ServiceWorkerState\n   open var onstatechange: ((Event) -> dynamic)?\n   fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)\n}\n\n/**\n * Exposes the JavaScript [ServiceWorkerRegistration](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerRegistration) to Kotlin\n */\npublic external abstract class ServiceWorkerRegistration : EventTarget {\n   open val installing: ServiceWorker?\n   open val waiting: ServiceWorker?\n   open val active: ServiceWorker?\n   open val scope: String\n   open var onupdatefound: ((Event) -> dynamic)?\n   open val APISpace: dynamic\n   fun update(): Promise<Unit>\n   fun unregister(): Promise<Boolean>\n   fun showNotification(title: String, options: NotificationOptions = definedExternally): Promise<Unit>\n   fun getNotifications(filter: GetNotificationOptions = definedExternally): Promise<Array<Notification>>\n   fun methodName(): Promise<dynamic>\n}\n\n/**\n * Exposes the JavaScript [ServiceWorkerContainer](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerContainer) to Kotlin\n */\npublic external abstract class ServiceWorkerContainer : EventTarget {\n   open val controller: ServiceWorker?\n   open val ready: Promise<ServiceWorkerRegistration>\n   open var oncontrollerchange: ((Event) -> dynamic)?\n   open var onmessage: ((MessageEvent) -> dynamic)?\n   fun register(scriptURL: String, options: RegistrationOptions = definedExternally): Promise<ServiceWorkerRegistration>\n   fun getRegistration(clientURL: String = definedExternally): Promise<Any?>\n   fun getRegistrations(): Promise<Array<ServiceWorkerRegistration>>\n   fun startMessages()\n}\n\npublic external interface RegistrationOptions {\n   var scope: String?\n   get() = definedExternally\n   set(value) = definedExternally\n   var type: WorkerType? /* = WorkerType.CLASSIC */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun RegistrationOptions(scope: String? = undefined, type: WorkerType? = WorkerType.CLASSIC): RegistrationOptions {\n   val o = js(\"({})\")\n   o[\"scope\"] = scope\n   o[\"type\"] = type\n   return o\n}\n\n/**\n * Exposes the JavaScript [ServiceWorkerMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerMessageEvent) to Kotlin\n */\npublic external open class ServiceWorkerMessageEvent(type: String, eventInitDict: ServiceWorkerMessageEventInit = definedExternally) : Event {\n   open val data: Any?\n   open val origin: String\n   open val lastEventId: String\n   open val source: UnionMessagePortOrServiceWorker?\n   open val ports: Array<out MessagePort>?\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface ServiceWorkerMessageEventInit : EventInit {\n   var data: Any?\n   get() = definedExternally\n   set(value) = definedExternally\n   var origin: String?\n   get() = definedExternally\n   set(value) = definedExternally\n   var lastEventId: String?\n   get() = definedExternally\n   set(value) = definedExternally\n   var source: UnionMessagePortOrServiceWorker?\n   get() = definedExternally\n   set(value) = definedExternally\n   var ports: Array<MessagePort>?\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ServiceWorkerMessageEventInit(data: Any? = undefined, origin: String? = undefined, lastEventId: String? = undefined, source: UnionMessagePortOrServiceWorker? = undefined, ports: Array<MessagePort>? = undefined, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): ServiceWorkerMessageEventInit {\n   val o = js(\"({})\")\n   o[\"data\"] = data\n   o[\"origin\"] = origin\n   o[\"lastEventId\"] = lastEventId\n   o[\"source\"] = source\n   o[\"ports\"] = ports\n   o[\"bubbles\"] = bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\n/**\n * Exposes the JavaScript [ServiceWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerGlobalScope) to Kotlin\n */\npublic external abstract class ServiceWorkerGlobalScope : WorkerGlobalScope {\n   open val clients: Clients\n   open val registration: ServiceWorkerRegistration\n   open var oninstall: ((Event) -> dynamic)?\n   open var onactivate: ((Event) -> dynamic)?\n   open var onfetch: ((FetchEvent) -> dynamic)?\n   open var

onforeignfetch: ((Event) -> dynamic)?\n   open var onmessage: ((MessageEvent) -> dynamic)?\n   open var onnotificationclick: ((NotificationEvent) -> dynamic)?\n   open var onnotificationclose: ((NotificationEvent) -> dynamic)?\n   open var onfunctionalevent: ((Event) -> dynamic)?\n   fun skipWaiting(): Promise<Unit>\n}\n\n/**\n * Exposes the JavaScript [Client](https://developer.mozilla.org/en/docs/Web/API/Client) to Kotlin\n */\npublic external abstract class Client : UnionClientOrMessagePortOrServiceWorker {\n   open val url: String\n   open val frameType: FrameType\n   open val id: String\n   fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)\n}\n\n/**\n * Exposes the JavaScript [WindowClient](https://developer.mozilla.org/en/docs/Web/API/WindowClient) to Kotlin\n */\npublic external abstract class WindowClient : Client {\n   open val visibilityState: dynamic\n   open val focused: Boolean\n   fun focus(): Promise<WindowClient>\n   fun navigate(url: String): Promise<WindowClient>\n}\n\n/**\n * Exposes the JavaScript [Clients](https://developer.mozilla.org/en/docs/Web/API/Clients) to Kotlin\n */\npublic external abstract class Clients {\n   fun get(id: String): Promise<Any?>\n   fun matchAll(options: ClientQueryOptions = definedExternally): Promise<Array<Client>>\n   fun openWindow(url: String): Promise<WindowClient?>\n   fun claim(): Promise<Unit>\n}\n\npublic external interface ClientQueryOptions {\n   var includeUncontrolled: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var type: ClientType? /* = ClientType.WINDOW */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ClientQueryOptions(includeUncontrolled: Boolean? = false, type: ClientType? = ClientType.WINDOW): ClientQueryOptions {\n   val o = js(\"({})\")\n   o[\"includeUncontrolled\"] = includeUncontrolled\n   o[\"type\"] = type\n   return o\n}\n\n/**\n * Exposes the JavaScript [ExtendableEvent](https://developer.mozilla.org/en/docs/Web/API/ExtendableEvent) to Kotlin\n */\npublic external open class ExtendableEvent(type: String, eventInitDict: ExtendableEventInit = definedExternally) : Event {\n   fun waitUntil(f: Promise<Any?>)\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface ExtendableEventInit : EventInit\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ExtendableEventInit(bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): ExtendableEventInit {\n   val o = js(\"({})\")\n   o[\"bubbles\"] = bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\n/**\n * Exposes the JavaScript [InstallEvent](https://developer.mozilla.org/en/docs/Web/API/InstallEvent) to Kotlin\n */\npublic external open class InstallEvent(type: String, eventInitDict: ExtendableEventInit = definedExternally) : ExtendableEvent {\n   fun registerForeignFetch(options: ForeignFetchOptions)\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface ForeignFetchOptions {\n   var scopes: Array<String>?\n   var origins: Array<String>?\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ForeignFetchOptions(scopes: Array<String>?, origins: Array<String>?): ForeignFetchOptions {\n   val o = js(\"({})\")\n   o[\"scopes\"] = scopes\n   o[\"origins\"] = origins\n   return o\n}\n\n/**\n * Exposes the JavaScript [FetchEvent](https://developer.mozilla.org/en/docs/Web/API/FetchEvent) to Kotlin\n */\npublic external open class FetchEvent(type: String, eventInitDict: FetchEventInit) : ExtendableEvent {\n   open val request: Request\n   open val clientId: String?\n   open val isReload: Boolean\n   fun respondWith(r: Promise<Response>)\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface FetchEventInit : ExtendableEventInit {\n   var request: Request?\n   var clientId: String? /* = null */\n   get() = definedExternally\n   set(value) = definedExternally\n   var isReload: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\",

\"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun FetchEventInit(request: Request?, clientId: String? = null, isReload: Boolean? = false, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): FetchEventInit {\n val o = js(\"({})\")\n o[\"request\"] = request\n o[\"clientId\"] = clientId\n o[\"isReload\"] = isReload\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\npublic external open class ForeignFetchEvent(type: String, eventInitDict: ForeignFetchEventInit) : ExtendableEvent {\n open val request: Request\n open val origin: String\n fun respondWith(r: Promise<ForeignFetchResponse>)\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface ForeignFetchEventInit : ExtendableEventInit {\n var request: Request?\n var origin: String? /* = \"null\" */\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ForeignFetchEventInit(request: Request?, origin: String? = \"null\", bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): ForeignFetchEventInit {\n val o = js(\"({})\")\n o[\"request\"] = request\n o[\"origin\"] = origin\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\npublic external interface ForeignFetchResponse {\n var response: Response?\n var origin: String?\n get() = definedExternally\n set(value) = definedExternally\n var headers: Array<String>?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ForeignFetchResponse(response: Response?, origin: String? = undefined, headers: Array<String>? = undefined): ForeignFetchResponse {\n val o = js(\"({})\")\n o[\"response\"] = response\n o[\"origin\"] = origin\n o[\"headers\"] = headers\n return o\n}\n\n/**\n * Exposes the JavaScript [ExtendableMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ExtendableMessageEvent) to Kotlin\n */\npublic external open class ExtendableMessageEvent(type: String, eventInitDict: ExtendableMessageEventInit = definedExternally) : ExtendableEvent {\n open val data: Any?\n open val origin: String\n open val lastEventId: String\n open val source: UnionClientOrMessagePortOrServiceWorker?\n open val ports: Array<out MessagePort>?\n\n companion object {\n val NONE: Short\n val CAPTURING_PHASE: Short\n val AT_TARGET: Short\n val BUBBLING_PHASE: Short\n }\n}\n\npublic external interface ExtendableMessageEventInit : ExtendableEventInit {\n var data: Any?\n get() = definedExternally\n set(value) = definedExternally\n var origin: String?\n get() = definedExternally\n set(value) = definedExternally\n var lastEventId: String?\n get() = definedExternally\n set(value) = definedExternally\n var source: UnionClientOrMessagePortOrServiceWorker?\n get() = definedExternally\n set(value) = definedExternally\n var ports: Array<MessagePort>?\n get() = definedExternally\n set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ExtendableMessageEventInit(data: Any? = undefined, origin: String? = undefined, lastEventId: String? = undefined, source: UnionClientOrMessagePortOrServiceWorker? = undefined, ports: Array<MessagePort>? = undefined, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): ExtendableMessageEventInit {\n val o = js(\"({})\")\n o[\"data\"] = data\n o[\"origin\"] = origin\n o[\"lastEventId\"] = lastEventId\n o[\"source\"] = source\n o[\"ports\"] = ports\n o[\"bubbles\"] = bubbles\n o[\"cancelable\"] = cancelable\n o[\"composed\"] = composed\n return o\n}\n\n/**\n * Exposes the JavaScript [Cache](https://developer.mozilla.org/en/docs/Web/API/Cache) to Kotlin\n */\npublic external abstract class Cache {\n fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>\n fun matchAll(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally): Promise<Array<Response>>\n fun add(request: dynamic): Promise<Unit>\n fun addAll(requests: Array<dynamic>): Promise<Unit>\n fun put(request: dynamic, response: Response): Promise<Unit>\n fun delete(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Boolean>\n fun keys(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally):

```
Promise<Array<Request>>\n}\n\npublic external interface CacheQueryOptions {\n    var ignoreSearch: Boolean? /* = false */\n        get() = definedExternally\n        set(value) = definedExternally\n    var ignoreMethod: Boolean? /* = false */\n        get() = definedExternally\n        set(value) = definedExternally\n    var ignoreVary: Boolean? /* = false */\n        get() = definedExternally\n        set(value) = definedExternally\n    var cacheName: String?\n        get() = definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun CacheQueryOptions(ignoreSearch: Boolean? = false, ignoreMethod: Boolean? = false, ignoreVary: Boolean? = false, cacheName: String? = undefined): CacheQueryOptions {\n    val o = js(\"({})\")\n    o[\"ignoreSearch\"] = ignoreSearch\n    o[\"ignoreMethod\"] = ignoreMethod\n    o[\"ignoreVary\"] = ignoreVary\n    o[\"cacheName\"] = cacheName\n    return o\n}\n\npublic external interface CacheBatchOperation {\n    var type: String?\n        get() = definedExternally\n        set(value) = definedExternally\n    var request: Request?\n        get() = definedExternally\n        set(value) = definedExternally\n    var response: Response?\n        get() = definedExternally\n        set(value) = definedExternally\n    var options: CacheQueryOptions?\n        get() = definedExternally\n        set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun CacheBatchOperation(type: String? = undefined, request: Request? = undefined, response: Response? = undefined, options: CacheQueryOptions? = undefined): CacheBatchOperation {\n    val o = js(\"({})\")\n    o[\"type\"] = type\n    o[\"request\"] = request\n    o[\"response\"] = response\n    o[\"options\"] = options\n    return o\n}\n\n/**\n * Exposes the JavaScript [CacheStorage](https://developer.mozilla.org/en/docs/Web/API/CacheStorage) to Kotlin\n */\npublic external abstract class CacheStorage {\n    fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>\n    fun has(cacheName: String): Promise<Boolean>\n    fun open(cacheName: String): Promise<Cache>\n    fun delete(cacheName: String): Promise<Boolean>\n    fun keys(): Promise<Array<String>>\n}\n\npublic external open class FunctionalEvent : ExtendableEvent {\n    companion object {\n        val NONE: Short\n        val CAPTURING_PHASE: Short\n        val AT_TARGET: Short\n        val BUBBLING_PHASE: Short\n    }\n}\n\npublic external interface UnionMessagePortOrServiceWorker\n\npublic external interface UnionClientOrMessagePortOrServiceWorker\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface ServiceWorkerState {\n    companion object\n}\n\npublic inline val ServiceWorkerState.Companion.INSTALLING: ServiceWorkerState get() = \"installing\".asDynamic().unsafeCast<ServiceWorkerState>()\n\npublic inline val ServiceWorkerState.Companion.INSTALLED: ServiceWorkerState get() = \"installed\".asDynamic().unsafeCast<ServiceWorkerState>()\n\npublic inline val ServiceWorkerState.Companion.ACTIVATING: ServiceWorkerState get() = \"activating\".asDynamic().unsafeCast<ServiceWorkerState>()\n\npublic inline val ServiceWorkerState.Companion.ACTIVATED: ServiceWorkerState get() = \"activated\".asDynamic().unsafeCast<ServiceWorkerState>()\n\npublic inline val ServiceWorkerState.Companion.REDUNDANT: ServiceWorkerState get() = \"redundant\".asDynamic().unsafeCast<ServiceWorkerState>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface FrameType {\n    companion object\n}\n\npublic inline val FrameType.Companion.AUXILIARY: FrameType get() = \"auxiliary\".asDynamic().unsafeCast<FrameType>()\n\npublic inline val FrameType.Companion.TOP_LEVEL: FrameType get() = \"top-level\".asDynamic().unsafeCast<FrameType>()\n\npublic inline val FrameType.Companion.NESTED: FrameType get() = \"nested\".asDynamic().unsafeCast<FrameType>()\n\npublic inline val FrameType.Companion.NONE: FrameType get() = \"none\".asDynamic().unsafeCast<FrameType>()\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic external interface ClientType {\n    companion object\n}\n\npublic inline val ClientType.Companion.WINDOW: ClientType get() = \"window\".asDynamic().unsafeCast<ClientType>()\n\npublic inline val ClientType.Companion.WORKER:
```

ClientType get() = \"worker\".asDynamic().unsafeCast<ClientType>()\n\npublic inline val ClientType.Companion.SHAREDWORKER: ClientType get() = \"sharedworker\".asDynamic().unsafeCast<ClientType>()\n\npublic inline val ClientType.Companion.ALL: ClientType get() = \"all\".asDynamic().unsafeCast<ClientType>()","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n\npackage org.w3c.xhr\n\nimport kotlin.js.*\nimport org.khronos.webgl.*\nimport org.w3c.dom.*\nimport org.w3c.dom.events.*\nimport org.w3c.files.*\n\n/**\n * Exposes the JavaScript [XMLHttpRequestEventTarget](https://developer.mozilla.org/en/docs/Web/API/XMLHttpRequestEventTarget) to Kotlin\n */\npublic external abstract class XMLHttpRequestEventTarget : EventTarget {\n   open var onloadstart: ((ProgressEvent) -> dynamic)?\n   open var onprogress: ((ProgressEvent) -> dynamic)?\n   open var onabort: ((Event) -> dynamic)?\n   open var onerror: ((Event) -> dynamic)?\n   open var onload: ((Event) -> dynamic)?\n   open var ontimeout: ((Event) -> dynamic)?\n   open var onloadend: ((Event) -> dynamic)?\n}\n\npublic external abstract class XMLHttpRequestUpload : XMLHttpRequestEventTarget\n\n/**\n * Exposes the JavaScript [XMLHttpRequest](https://developer.mozilla.org/en/docs/Web/API/XMLHttpRequest) to Kotlin\n */\npublic external open class XMLHttpRequest : XMLHttpRequestEventTarget {\n   var onreadystatechange: ((Event) -> dynamic)?\n   open val readyState: Short\n   var timeout: Int\n   var withCredentials: Boolean\n   open val upload: XMLHttpRequestUpload\n   open val responseURL: String\n   open val status: Short\n   open val statusText: String\n   var responseType: XMLHttpRequestResponseType\n   open val response: Any?\n   open val responseText: String\n   open val responseXML: Document?\n   fun open(method: String, url: String)\n   fun open(method: String, url: String, async: Boolean, username: String? = definedExternally, password: String? = definedExternally)\n   fun setRequestHeader(name: String, value: String)\n   fun send(body: dynamic = definedExternally)\n   fun abort()\n   fun getResponseHeader(name: String): String?\n   fun getAllResponseHeaders(): String\n   fun overrideMimeType(mime: String)\n\n   companion object {\n   val UNSENT: Short\n   val OPENED: Short\n   val HEADERS_RECEIVED: Short\n   val LOADING: Short\n   val DONE: Short\n   }\n}\n\n/**\n * Exposes the JavaScript [FormData](https://developer.mozilla.org/en/docs/Web/API/FormData) to Kotlin\n */\npublic external open class FormData(form: HTMLFormElement = definedExternally) {\n   fun append(name: String, value: String)\n   fun append(name: String, value: Blob, filename: String = definedExternally)\n   fun delete(name: String)\n   fun get(name: String): dynamic\n   fun getAll(name: String): Array<dynamic>\n   fun has(name: String): Boolean\n   fun set(name: String, value: String)\n   fun set(name: String, value: Blob, filename: String = definedExternally)\n}\n\n/**\n * Exposes the JavaScript [ProgressEvent](https://developer.mozilla.org/en/docs/Web/API/ProgressEvent) to Kotlin\n */\npublic external open class ProgressEvent(type: String, eventInitDict: ProgressEventInit = definedExternally) : Event {\n   open val lengthComputable: Boolean\n   open val loaded: Number\n   open val total: Number\n\n   companion object {\n   val NONE: Short\n   val CAPTURING_PHASE: Short\n   val AT_TARGET: Short\n   val BUBBLING_PHASE: Short\n   }\n}\n\npublic external interface ProgressEventInit : EventInit {\n   var lengthComputable: Boolean? /* = false */\n   get() = definedExternally\n   set(value) = definedExternally\n   var loaded: Number? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n   var total: Number? /* = 0 */\n   get() = definedExternally\n   set(value) = definedExternally\n}\n\n@Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\n@kotlin.internal.InlineOnly\npublic inline fun ProgressEventInit(lengthComputable: Boolean? = false, loaded: Number? = 0, total: Number? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): ProgressEventInit {\n   val o = js(\"({})\")\n   o[\"lengthComputable\"] = lengthComputable\n   o[\"loaded\"] = loaded\n   o[\"total\"] = total\n   o[\"bubbles\"] = bubbles\n   o[\"cancelable\"] = cancelable\n   o[\"composed\"] = composed\n   return o\n}\n\n/* please, don't implement this interface! */\n@JsName(\"null\")\n@Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\npublic

external interface XMLHttpRequestResponseType {\n   companion object\n}\n\npublic inline val XMLHttpRequestResponseType.Companion.EMPTY: XMLHttpRequestResponseType get() = \"\".asDynamic().unsafeCast<XMLHttpRequestResponseType>()\n\npublic inline val XMLHttpRequestResponseType.Companion.ARRAYBUFFER: XMLHttpRequestResponseType get() = \"arraybuffer\".asDynamic().unsafeCast<XMLHttpRequestResponseType>()\n\npublic inline val XMLHttpRequestResponseType.Companion.BLOB: XMLHttpRequestResponseType get() = \"blob\".asDynamic().unsafeCast<XMLHttpRequestResponseType>()\n\npublic inline val XMLHttpRequestResponseType.Companion.DOCUMENT: XMLHttpRequestResponseType get() = \"document\".asDynamic().unsafeCast<XMLHttpRequestResponseType>()\n\npublic inline val XMLHttpRequestResponseType.Companion.JSON: XMLHttpRequestResponseType get() = \"json\".asDynamic().unsafeCast<XMLHttpRequestResponseType>()\n\npublic inline val XMLHttpRequestResponseType.Companion.TEXT: XMLHttpRequestResponseType get() = \"text\".asDynamic().unsafeCast<XMLHttpRequestResponseType>()","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\nimport kotlin.annotation.AnnotationRetention.BINARY\nimport kotlin.annotation.AnnotationRetention.SOURCE\nimport kotlin.annotation.AnnotationTarget.*\nimport kotlin.internal.RequireKotlin\nimport kotlin.internal.RequireKotlinVersionKind\nimport kotlin.reflect.KClass\n\n/**\n * Signals that the annotated annotation class is a marker of an experimental API.\n *\n * Any declaration annotated with that marker is considered an experimental declaration\n * and its call sites should accept the experimental aspect of it either by using [UseExperimental],\n * or by being annotated with that marker themselves, effectively causing further propagation of that experimental aspect.\n *\n * This class is deprecated in favor of a more general approach provided by [RequiresOptIn]/[OptIn].\n */\n@Target(ANNOTATION_CLASS)\n@Retention(BINARY)\n@SinceKotlin(\"1.2\")\n@RequireKotlin(\"1.2.50\", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\n@Deprecated(\"Please use RequiresOptIn instead.\")\npublic annotation class Experimental(val level: Level = Level.ERROR) {\n   /**\n   * Severity of the diagnostic that should be reported on usages of experimental API which did not explicitly accept the experimental aspect\n   * of that API either by using [UseExperimental] or by being annotated with the corresponding marker annotation.\n   */\n   public enum class Level {\n   /** Specifies that a warning should be reported on incorrect usages of this experimental API. */\n   WARNING,\n   /** Specifies that an error should be reported on incorrect usages of this experimental API. */\n   ERROR,\n   }\n}\n\n/**\n * Allows to use experimental API denoted by the given markers in the annotated file, declaration, or expression.\n * If a declaration is annotated with [UseExperimental], its usages are **not** required to opt-in to that experimental API.\n *\n * This class is deprecated in favor of a more general approach provided by [RequiresOptIn]/[OptIn].\n */\n@Target(\n   CLASS, PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE,\n   TYPEALIAS\n)\n@Retention(SOURCE)\n@SinceKotlin(\"1.2\")\n@RequireKotlin(\"1.2.50\", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)\n@DeprecatedSinceKotlin(warningSince = \"1.4\", errorSince = \"1.6\")\n@Deprecated(\"Please use OptIn instead.\", ReplaceWith(\"OptIn(*markerClass)\", \"kotlin.OptIn\"))\npublic annotation class UseExperimental(\n   vararg val markerClass: KClass<out Annotation>\n)\n\n\n@Target(CLASS, PROPERTY, CONSTRUCTOR, FUNCTION, TYPEALIAS)\n@Retention(BINARY)\ninternal annotation class WasExperimental(\n   vararg val markerClass: KClass<out Annotation>\n)\n","package kotlin\n\nimport kotlin.annotation.AnnotationTarget.*\n\n/**\n * This annotation marks the standard library API that is considered experimental and is not subject to the\n * [general compatibility guarantees](https://kotlinlang.org/docs/reference/evolution/components-stability.html) given for the standard library:\n * the behavior of such API may be changed or the API may be removed completely in any further release.\n *\n * > Beware using the annotated API especially if you're developing a library, since your library

might become binary incompatible\n * with the future versions of the standard library.\n *\n * Any usage of a declaration annotated with `@ExperimentalStdlibApi` must be accepted either by\n * annotating that usage with the [OptIn] annotation, e.g. `@OptIn(ExperimentalStdlibApi::class)`,\n * or by using the compiler argument `-opt-in=kotlin.ExperimentalStdlibApi`.\n */\n@RequiresOptIn(level = RequiresOptIn.Level.ERROR)\n@Retention(AnnotationRetention.BINARY)\n@Target(\n CLASS,\n ANNOTATION_CLASS,\n PROPERTY,\n FIELD,\n LOCAL_VARIABLE,\n VALUE_PARAMETER,\n CONSTRUCTOR,\n FUNCTION,\n PROPERTY_GETTER,\n PROPERTY_SETTER,\n TYPEALIAS\n)\n@MustBeDocumented\n@SinceKotlin(\"1.3\")\npublic annotation class ExperimentalStdlibApi\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\nimport kotlin.annotation.AnnotationTarget.*\nimport kotlin.experimental.ExperimentalTypeInference\n\n/**\n * Allows to infer generic type arguments of a function from the calls in the annotated function parameter of that function.\n *\n * When this annotation is placed on a generic function parameter of a function,\n * it enables to infer the type arguments of that generic function from the lambda body passed to that parameter.\n *\n * The calls that affect inference are either members of the receiver type of an annotated function parameter or\n * extensions for that type. The extensions must be themselves annotated with `@BuilderInference`.\n *\n * Example: we declare\n * ```\n * fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T>\n * ```\n * and use it like\n * ```\n * val result = sequence { yield(\"result\") }\n * ```\n * Here the type argument of the resulting sequence is inferred to `String` from\n * the argument of the [SequenceScope.yield] function, that is called inside the lambda passed to [sequence].\n *\n * Note: this annotation is experimental, see [ExperimentalTypeInference] on how to opt-in for it.\n */\n@Target(VALUE_PARAMETER, FUNCTION, PROPERTY)\n@Retention(AnnotationRetention.BINARY)\n@SinceKotlin(\"1.3\")\n@ExperimentalTypeInference\npublic annotation class BuilderInference\n\n\n/**\n * Enables overload selection based on the type of the value returned from lambda argument.\n *\n * When two or more function overloads have otherwise the same parameter lists that differ only in the return type\n * of a functional parameter, this annotation enables overload selection by the type of the value returned from\n * the lambda function passed to this functional parameter.\n *\n * Example:\n * ```\n * @OverloadResolutionByLambdaReturnType\n * fun create(intProducer: () -> Int): Int\n *\n * fun create(doubleProducer: () -> Double): Double\n *\n * val newValue = create { 3.14 }\n * ```\n *\n * The annotation being applied to one of overloads allows to resolve this ambiguity by analyzing what value is returned\n * from the lambda function.\n *\n * This annotation is also used to discriminate the annotated overloads in case if overload selection still cannot\n * choose one of them even taking in account the result of lambda parameter analysis. In that case a warning is reported.\n *\n * Note: this annotation is experimental, see [ExperimentalTypeInference] on how to opt-in for it.\n */\n@Target(FUNCTION)\n@Retention(AnnotationRetention.BINARY)\n@SinceKotlin(\"1.4\")\n@ExperimentalTypeInference\npublic annotation class OverloadResolutionByLambdaReturnType","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\nimport kotlin.annotation.AnnotationTarget.*\nimport kotlin.internal.RequireKotlin\nimport kotlin.internal.RequireKotlinVersionKind\n\n/**\n * The experimental multiplatform support API marker.\n *\n * Any usage of a declaration annotated with `@ExperimentalMultiplatform` must be accepted either by\n * annotating that usage with the [OptIn] annotation, e.g. `@OptIn(ExperimentalMultiplatform::class)`,\n * or by using the compiler argument `-opt-in=kotlin.ExperimentalMultiplatform`.\n */\n@RequiresOptIn\n@MustBeDocumented\n@Target(\n CLASS,\n ANNOTATION_CLASS,\n PROPERTY,\n FIELD,\n LOCAL_VARIABLE,\n VALUE_PARAMETER,\n CONSTRUCTOR,\n FUNCTION,\n PROPERTY_GETTER,\n PROPERTY_SETTER,\n TYPEALIAS\n)\n@Retention(AnnotationRetention.BINARY)\n@RequireKotlin(\"1.2.50\", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)\npublic annotation class ExperimentalMultiplatform\n\n/**\n

* Marks an expected annotation class that it isn't required to have actual counterparts in all platforms.\n *\n * This annotation is only applicable to `expect` annotation classes in multi-platform projects and marks that class as \"optional\".\n * Optional expected class is allowed to have no corresponding actual class on the platform. Optional annotations can only be used\n * to annotate something, not as types in signatures. If an optional annotation has no corresponding actual class on a platform,\n * the annotation entries where it's used are simply erased when compiling code on that platform.\n *\n * Note: this annotation is experimental, see [ExperimentalMultiplatform] on how to opt-in for it.\n */\n@Target(ANNOTATION_CLASS)\n@Retention(AnnotationRetention.BINARY)\n@ExperimentalMultiplatform\n@RequireKotlin(\"1.2.50\", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)\npublic annotation class OptionalExpectation\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\nimport kotlin.annotation.AnnotationRetention.BINARY\nimport kotlin.annotation.AnnotationRetention.SOURCE\nimport kotlin.annotation.AnnotationTarget.*\nimport kotlin.internal.RequireKotlin\nimport kotlin.internal.RequireKotlinVersionKind\nimport kotlin.reflect.KClass\n\n/**\n * Signals that the annotated annotation class is a marker of an API that requires an explicit opt-in.\n *\n * Call sites of any declaration annotated with that marker should opt in to the API either by using [OptIn],\n * or by being annotated with that marker themselves, effectively causing further propagation of the opt-in requirement.\n *\n * This class requires opt-in itself and can only be used with the compiler argument `-opt-in=kotlin.RequiresOptIn`.\n *\n * @property message message to be reported on usages of API without an explicit opt-in, or empty string for the default message.\n *
The default message is: \"This declaration is experimental and its usage should be marked with 'Marker'\n *
or '@OptIn(Marker::class)'\", where `Marker` is the opt-in requirement marker.\n * @property level specifies how usages of API without an explicit opt-in are reported in code.\n */\n@Target(ANNOTATION_CLASS)\n@Retention(BINARY)\n@SinceKotlin(\"1.3\")\n@RequireKotlin(\"1.3.70\", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)\npublic annotation class RequiresOptIn(\n val message: String = \"\",\n val level: Level = Level.ERROR\n) {\n /**\n * Severity of the diagnostic that should be reported on usages which did not explicitly opted into\n * the API either by using [OptIn] or by being annotated with the corresponding marker annotation.\n */\n public enum class Level {\n /** Specifies that a warning should be reported on incorrect usages of this API. */\n WARNING,\n\n /** Specifies that an error should be reported on incorrect usages of this API. */\n ERROR,\n }\n}\n\n/**\n * Allows to use the API denoted by the given markers in the annotated file, declaration, or expression.\n * If a declaration is annotated with [OptIn], its usages are **not** required to opt in to that API.\n *\n * This class requires opt-in itself and can only be used with the compiler argument `-opt-in=kotlin.RequiresOptIn`.\n */\n@Target(\n CLASS, PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE, TYPEALIAS\n)\n@Retention(SOURCE)\n@SinceKotlin(\"1.3\")\n@RequireKotlin(\"1.3.70\", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)\npublic annotation class OptIn(\n vararg val markerClass: KClass<out Annotation>\n)\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\npackage kotlin.collections\n\nimport kotlin.js.JsName\n\n/**\n * Provides a skeletal implementation of the read-only [Collection] interface.\n *\n * @param E the type of elements contained in the collection. The collection is covariant in its element type.\n */\n@SinceKotlin(\"1.1\")\npublic abstract class AbstractCollection<out E> protected constructor() : Collection<E> {\n abstract override val size: Int\n abstract override fun iterator(): Iterator<E>\n\n override fun contains(element: @UnsafeVariance E): Boolean = any { it == element }\n\n override fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean =\n elements.all { contains(it) } // use when js will support bound refs: elements.all(this::contains)\n\n override fun isEmpty(): Boolean = size == 0\n\n override fun toString(): String = joinToString(\", \", \"[\", \"]\") {\n if (it === this) \"(this Collection)\" else it.toString()\n }\n\n /**\n * Returns new array of type `Array<Any?>` with

the elements of this collection.\n */\n @JsName(\"toArray\")\n protected open fun toArray(): Array<Any?> = copyToArrayImpl(this)\n\n /**\n * Fills the provided [array] or creates new array of the same type\n * and fills it with the elements of this collection.\n */\n protected open fun <T> toArray(array: Array<T>): Array<T> = copyToArrayImpl(this, array)\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n\nprivate enum class State {\n Ready,\n NotReady,\n Done,\n Failed\n}\n\n/**\n * A base class to simplify implementing iterators so that implementations only have to implement [computeNext]\n * to implement the iterator, calling [done] when the iteration is complete.\n */\npublic abstract class AbstractIterator<T> : Iterator<T> {\n private var state = State.NotReady\n private var nextValue: T? = null\n\n override fun hasNext(): Boolean {\n require(state != State.Failed)\n return when (state) {\n State.Done -> false\n State.Ready -> true\n else -> tryToComputeNext()\n }\n }\n\n override fun next(): T {\n if (!hasNext()) throw NoSuchElementException()\n state = State.NotReady\n @Suppress(\"UNCHECKED_CAST\")\n return nextValue as T\n }\n\n private fun tryToComputeNext(): Boolean {\n state = State.Failed\n computeNext()\n return state == State.Ready\n }\n\n /**\n * Computes the next item in the iterator.\n *\n * This callback method should call one of these two methods:\n *\n * * [setNext] with the next value of the iteration\n * * [done] to indicate there are no more elements\n *\n * Failure to call either method will result in the iteration terminating with a failed state\n */\n abstract protected fun computeNext(): Unit\n\n /**\n * Sets the next value in the iteration, called from the [computeNext] function\n */\n protected fun setNext(value: T): Unit {\n nextValue = value\n state = State.Ready\n }\n\n /**\n * Sets the state to done so that the iteration terminates.\n */\n protected fun done() {\n state = State.Done\n }\n}\n\n\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n/*\n * Based on GWT AbstractList\n * Copyright 2007 Google Inc.\n*/\n\npackage kotlin.collections\n\n/**\n * Provides a skeletal implementation of the read-only [List] interface.\n *\n * This class is intended to help implementing read-only lists so it doesn't support concurrent modification tracking.\n *\n * @param E the type of elements contained in the list. The list is covariant in its element type.\n */\n@SinceKotlin(\"1.1\")\npublic abstract class AbstractList<out E> protected constructor() : AbstractCollection<E>(), List<E> {\n abstract override val size: Int\n abstract override fun get(index: Int): E\n\n override fun iterator(): Iterator<E> = IteratorImpl()\n\n override fun indexOf(element: @UnsafeVariance E): Int = indexOfFirst { it == element }\n\n override fun lastIndexOf(element: @UnsafeVariance E): Int = indexOfLast { it == element }\n\n override fun listIterator(): ListIterator<E> = ListIteratorImpl(0)\n\n override fun listIterator(index: Int): ListIterator<E> = ListIteratorImpl(index)\n\n override fun subList(fromIndex: Int, toIndex: Int): List<E> = SubList(this, fromIndex, toIndex)\n\n private class SubList<out E>(private val list: AbstractList<E>, private val fromIndex: Int, toIndex: Int) : AbstractList<E>(), RandomAccess {\n private var _size: Int = 0\n\n init {\n checkRangeIndexes(fromIndex, toIndex, list.size)\n this._size = toIndex - fromIndex\n }\n\n override fun get(index: Int): E {\n checkElementIndex(index, _size)\n\n return list[fromIndex + index]\n }\n\n override val size: Int get() = _size\n }\n\n /**\n * Compares this list with other list instance with the ordered structural equality.\n *\n * @return true, if [other] instance is a [List] of the same size, which contains the same elements in the same order.\n */\n override fun equals(other: Any?): Boolean {\n if (other === this) return true\n if (other !is List<*>) return false\n\n return orderedEquals(this, other)\n }\n\n /**\n * Returns the hash code value for this list.\n */\n override fun hashCode(): Int = orderedHashCode(this)\n\n private open inner class IteratorImpl : Iterator<E> {\n /** the index of the item that will be returned on the next call to [next]`()` */\n protected var index = 0\n\n override fun hasNext(): Boolean = index < size\n\n override fun next(): E {\n if (!hasNext()) throw NoSuchElementException()\n return get(index++)\n }\n }\n\n /**\n * Implementation of [ListIterator] for abstract lists.\n */\n private open inner class ListIteratorImpl(index: Int) : IteratorImpl(), ListIterator<E> {\n\n init {\n checkPositionIndex(index, this@AbstractList.size)\n this.index = index\n }\n\n override fun hasPrevious(): Boolean = index > 0\n\n override fun

```
nextIndex(): Int = index\n\n        override fun previous(): E {\n            if (!hasPrevious()) throw NoSuchElementException()\n            return get(--index)\n        }\n\n        override fun previousIndex(): Int = index - 1\n    }\n\n    internal companion object {\n        internal fun checkElementIndex(index: Int, size: Int) {\n            if (index < 0 || index >= size) {\n                throw IndexOutOfBoundsException(\"index: $index, size: $size\")\n            }\n        }\n\n        internal fun checkPositionIndex(index: Int, size: Int) {\n            if (index < 0 || index > size) {\n                throw IndexOutOfBoundsException(\"index: $index, size: $size\")\n            }\n        }\n\n        internal fun checkRangeIndexes(fromIndex: Int, toIndex: Int, size: Int) {\n            if (fromIndex < 0 || toIndex > size) {\n                throw IndexOutOfBoundsException(\"fromIndex: $fromIndex, toIndex: $toIndex, size: $size\")\n            }\n            if (fromIndex > toIndex) {\n                throw IllegalArgumentException(\"fromIndex: $fromIndex > toIndex: $toIndex\")\n            }\n        }\n\n        internal fun checkBoundsIndexes(startIndex: Int, endIndex: Int, size: Int) {\n            if (startIndex < 0 || endIndex > size) {\n                throw IndexOutOfBoundsException(\"startIndex: $startIndex, endIndex: $endIndex, size: $size\")\n            }\n            if (startIndex > endIndex) {\n                throw IllegalArgumentException(\"startIndex: $startIndex > endIndex: $endIndex\")\n            }\n        }\n\n        internal fun orderedHashCode(c: Collection<*>): Int {\n            var hashCode = 1\n            for (e in c) {\n                hashCode = 31 * hashCode + (e?.hashCode() ?: 0)\n            }\n            return hashCode\n        }\n\n        internal fun orderedEquals(c: Collection<*>, other: Collection<*>): Boolean {\n            if (c.size != other.size) return false\n\n            val otherIterator = other.iterator()\n            for (elem in c) {\n                val elemOther = otherIterator.next()\n                if (elem != elemOther) {\n                    return false\n                }\n            }\n            return true\n        }\n    }\n}","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n/*\n * Based on GWT AbstractMap\n * Copyright 2007 Google Inc.\n */\n\npackage kotlin.collections\n\n/**\n * Provides a skeletal implementation of the read-only [Map] interface.\n *\n * The implementor is required to implement [entries] property, which should return read-only set of map entries.\n *\n * @param K the type of map keys. The map is invariant in its key type.\n * @param V the type of map values. The map is covariant in its value type.\n */\n@SinceKotlin(\"1.1\")\npublic abstract class AbstractMap<K, out V> protected constructor() : Map<K, V> {\n\n    override fun containsKey(key: K): Boolean {\n        return implFindEntry(key) != null\n    }\n\n    override fun containsValue(value: @UnsafeVariance V): Boolean = entries.any { it.value == value }\n\n    internal fun containsEntry(entry: Map.Entry<*, *>?): Boolean {\n        // since entry comes from @UnsafeVariance parameters it can be virtually anything\n        if (entry !is Map.Entry<*, *>) return false\n        val key = entry.key\n        val value = entry.value\n        val ourValue = get(key)\n\n        if (value != ourValue) {\n            return false\n        }\n\n        // Perhaps it was null and we don't contain the key?\n        if (ourValue == null && !containsKey(key)) {\n            return false\n        }\n\n        return true\n    }\n\n\n    /**\n     * Compares this map with other instance with the ordered structural equality.\n     *\n     * @return true, if [other] instance is a [Map] of the same size, all entries of which are contained in the [entries] set of this map.\n     */\n    override fun equals(other: Any?): Boolean {\n        if (other === this) return true\n        if (other !is Map<*, *>) return false\n        if (size != other.size) return false\n\n        return other.entries.all { containsEntry(it) }\n    }\n\n    override operator fun get(key: K): V? = implFindEntry(key)?.value\n\n\n    /**\n     * Returns the hash code value for this map.\n     *\n     * It is the same as the hashCode of [entries] set.\n     */\n    override fun hashCode(): Int = entries.hashCode()\n\n    override fun isEmpty(): Boolean = size == 0\n    override val size: Int get() = entries.size\n\n    /**\n     * Returns a read-only [Set] of all keys in this map.\n     *\n     * Accessing this property first time creates a keys view from [entries].\n     * All subsequent accesses just return the created instance.\n     */\n    override val keys: Set<K>\n        get() {\n            if (_keys == null) {\n                _keys = object : AbstractSet<K>() {\n                    override operator fun contains(element: K): Boolean = containsKey(element)\n\n                    override operator fun iterator(): Iterator<K> {\n                        val entryIterator = entries.iterator()\n                        return object : Iterator<K> {\n                            override fun hasNext(): Boolean = entryIterator.hasNext()\n                            override fun next(): K = entryIterator.next().key\n                        }\n                    }\n\n                    override val size: Int get() = this@AbstractMap.size\n                }\n            }\n            return _keys!!\n        }\n\n    @kotlin.jvm.Volatile\n    private var _keys: Set<K>? = null\n\n\n    override fun toString(): String = entries.joinToString(\", \", \"{\", \"}\") {
```

toString(it) }\n\n   private fun toString(entry: Map.Entry<K, V>): String = toString(entry.key) + \"=\" + toString(entry.value)\n\n   private fun toString(o: Any?): String = if (o === this) \"(this Map)\" else o.toString()\n\n   /**\n   * Returns a read-only [Collection] of all values in this map.\n   *\n   * Accessing this property first time creates a values view from [entries].\n   * All subsequent accesses just return the created instance.\n   */\n   override val values: Collection<V>\n   get() {\n   if (_values == null) {\n   _values = object : AbstractCollection<V>() {\n   override operator fun contains(element: @UnsafeVariance V): Boolean = containsValue(element)\n\n   override operator fun iterator(): Iterator<V> {\n   val entryIterator = entries.iterator()\n   return object : Iterator<V> {\n   override fun hasNext(): Boolean = entryIterator.hasNext()\n   override fun next(): V = entryIterator.next().value\n   }\n   }\n\n   override val size: Int get() = this@AbstractMap.size\n   }\n   }\n   return _values!!\n   }\n\n   @kotlin.jvm.Volatile\n   private var _values: Collection<V>? = null\n\n   private fun implFindEntry(key: K): Map.Entry<K, V>? = entries.firstOrNull { it.key == key }\n\n   internal companion object {\n\n   internal fun entryHashCode(e: Map.Entry<*, *>): Int = with(e) { (key?.hashCode() ?: 0) xor (value?.hashCode() ?: 0) }\n   internal fun entryToString(e: Map.Entry<*, *>): String = with(e) { \"$key=$value\" }\n   internal fun entryEquals(e: Map.Entry<*, *>, other: Any?): Boolean {\n   if (other !is Map.Entry<*, *>) return false\n   return e.key == other.key && e.value == other.value\n   }\n   }\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\npackage kotlin.collections\n\n/**\n * Provides a skeletal implementation of the read-only [Set] interface.\n *\n * This class is intended to help implementing read-only sets so it doesn't support concurrent modification tracking.\n *\n * @param E the type of elements contained in the set. The set is covariant in its element type.\n */\n@SinceKotlin(\"1.1\")\npublic abstract class AbstractSet<out E> protected constructor() : AbstractCollection<E>(), Set<E> {\n\n   /**\n   * Compares this set with other set instance with the unordered structural equality.\n   *\n   * @return true, if [other] instance is a [Set] of the same size, all elements of which are contained in this set.\n   */\n   override fun equals(other: Any?): Boolean {\n   if (other === this) return true\n   if (other !is Set<*>) return false\n   return setEquals(this, other)\n   }\n\n   /**\n   * Returns the hash code value for this set.\n   */\n   override fun hashCode(): Int = unorderedHashCode(this)\n\n   internal companion object {\n   internal fun unorderedHashCode(c: Collection<*>): Int {\n   var hashCode = 0\n   for (element in c) {\n   hashCode += (element?.hashCode() ?: 0)\n   }\n   return hashCode\n   }\n\n   internal fun setEquals(c: Set<*>, other: Set<*>): Boolean {\n   if (c.size != other.size) return false\n   return c.containsAll(other)\n   }\n   }\n}","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n/**\n * Resizable-array implementation of the deque data structure.\n *\n * The name deque is short for \"double ended queue\" and is usually pronounced \"deck\".\n *\n * The collection provide methods for convenient access to the both ends.\n * It also implements [MutableList] interface and supports efficient get/set operations by index.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic class ArrayDeque<E> : AbstractMutableList<E> {\n   private var head: Int = 0\n   private var elementData: Array<Any?>\n\n   override var size: Int = 0\n   private set\n\n   /**\n   * Constructs an empty deque with specified [initialCapacity], or throws [IllegalArgumentException] if [initialCapacity] is negative.\n   */\n   public constructor(initialCapacity: Int) {\n   elementData = when {\n   initialCapacity == 0 -> emptyElementData\n   initialCapacity > 0 -> arrayOfNulls(initialCapacity)\n   else -> throw IllegalArgumentException(\"Illegal Capacity: $initialCapacity\")\n   }\n   }\n\n   /**\n   * Constructs an empty deque.\n   */\n   public constructor() {\n   elementData = emptyElementData\n   }\n\n   /**\n   * Constructs a deque that contains the same elements as the specified [elements] collection in the same order.\n   */\n   public constructor(elements: Collection<E>) {\n   elementData = elements.toTypedArray()\n   size = elementData.size\n   if (elementData.isEmpty())\n   elementData = emptyElementData\n   }\n\n   /**\n   * Ensures that the capacity of this deque is at least equal to the specified [minCapacity].\n   *\n   * If the current capacity is less than the [minCapacity], a new backing

```
storage is allocated with greater capacity.\n     * Otherwise, this method takes no action and simply returns.\n     */\n
 private fun ensureCapacity(minCapacity: Int) {\n        if (minCapacity < 0) throw IllegalStateException(\"Deque is
too big.\")    // overflow\n        if (minCapacity <= elementData.size) return\n        if (elementData ===
emptyElementData) {\n            elementData = arrayOfNulls(minCapacity.coerceAtLeast(defaultMinCapacity))\n
    return\n        }\n\n        val newCapacity = newCapacity(elementData.size, minCapacity)\n
copyElements(newCapacity)\n    }\n\n    /**\n     * Creates a new array with the specified [newCapacity] size and
copies elements in the [elementData] array to it.\n     */\n    private fun copyElements(newCapacity: Int) {\n        val
newElements = arrayOfNulls<Any?>(newCapacity)\n        elementData.copyInto(newElements, 0, head,
elementData.size)\n        elementData.copyInto(newElements, elementData.size - head, 0, head)\n        head = 0\n
 elementData = newElements\n    }\n\n    @kotlin.internal.InlineOnly\n    private inline fun
internalGet(internalIndex: Int): E {\n        @Suppress(\"UNCHECKED_CAST\")\n        return
elementData[internalIndex] as E\n    }\n\n    private fun positiveMod(index: Int): Int = if (index >=
elementData.size) index - elementData.size else index\n\n    private fun negativeMod(index: Int): Int = if (index < 0)
index + elementData.size else index\n\n    @kotlin.internal.InlineOnly\n    private inline fun internalIndex(index:
Int): Int = positiveMod(head + index)\n\n    private fun incremented(index: Int): Int = if (index ==
elementData.lastIndex) 0 else index + 1\n\n    private fun decremented(index: Int): Int = if (index == 0)
elementData.lastIndex else index - 1\n\n    override fun isEmpty(): Boolean = size == 0\n\n    /**\n     * Returns the
first element, or throws [NoSuchElementException] if this deque is empty.\n     */\n    public fun first(): E = if
(isEmpty()) throw NoSuchElementException(\"ArrayDeque is empty.\") else internalGet(head)\n\n    /**\n     *
Returns the first element, or `null` if this deque is empty.\n     */\n    public fun firstOrNull(): E? = if (isEmpty()) null
else internalGet(head)\n\n    /**\n     * Returns the last element, or throws [NoSuchElementException] if this deque
is empty.\n     */\n    public fun last(): E = if (isEmpty()) throw NoSuchElementException(\"ArrayDeque is empty.\")
else internalGet(internalIndex(lastIndex))\n\n    /**\n     * Returns the last element, or `null` if this deque is empty.\n
   */\n    public fun lastOrNull(): E? = if (isEmpty()) null else internalGet(internalIndex(lastIndex))\n\n    /**\n     *
Prepends the specified [element] to this deque.\n     */\n    public fun addFirst(element: E) {\n
ensureCapacity(size + 1)\n\n        head = decremented(head)\n        elementData[head] = element\n        size += 1\n
}\n\n    /**\n     * Appends the specified [element] to this deque.\n     */\n    public fun addLast(element: E) {\n
ensureCapacity(size + 1)\n\n        elementData[internalIndex(size)] = element\n        size += 1\n    }\n\n    /**\n     *
Removes the first element from this deque and returns that removed element, or throws [NoSuchElementException]
if this deque is empty.\n     */\n    public fun removeFirst(): E {\n        if (isEmpty()) throw
NoSuchElementException(\"ArrayDeque is empty.\")\n\n        val element = internalGet(head)\n
elementData[head] = null\n        head = incremented(head)\n        size -= 1\n        return element\n    }\n\n    /**\n
* Removes the first element from this deque and returns that removed element, or returns `null` if this deque is
empty.\n     */\n    public fun removeFirstOrNull(): E? = if (isEmpty()) null else removeFirst()\n\n    /**\n     *
Removes the last element from this deque and returns that removed element, or throws [NoSuchElementException]
if this deque is empty.\n     */\n    public fun removeLast(): E {\n        if (isEmpty()) throw
NoSuchElementException(\"ArrayDeque is empty.\")\n\n        val internalLastIndex = internalIndex(lastIndex)\n
val element = internalGet(internalLastIndex)\n        elementData[internalLastIndex] = null\n        size -= 1\n
return element\n    }\n\n    /**\n     * Removes the last element from this deque and returns that removed element, or
returns `null` if this deque is empty.\n     */\n    public fun removeLastOrNull(): E? = if (isEmpty()) null else
removeLast()\n\n    // MutableList, MutableCollection\n    public override fun add(element: E): Boolean {\n
addLast(element)\n        return true\n    }\n\n    public override fun add(index: Int, element: E) {\n
AbstractList.checkPositionIndex(index, size)\n\n        if (index == size) {\n            addLast(element)\n
return\n        } else if (index == 0) {\n            addFirst(element)\n            return\n        }\n\n        ensureCapacity(size
+ 1)\n\n        // Elements in circular array lay in 2 ways:\n        //   1. `head` is less than `tail`:       [#, #, e1, e2, e3,
#]\n        //   2. `head` is greater than `tail`:    [e3, #, #, #, e1, e2]\n        // where head is the index of the first element
in the circular array,\n        // and tail is the index following the last element.\n        //\n        // At this point the
insertion index is not equal to head or tail.\n        // Also the circular array can store at least one more element.\n
```

```
//\n        // Depending on where the given element must be inserted the preceding or the succeeding\n        // elements will be shifted to make room for the element to be inserted.\n        //\n        // In case the preceding elements are shifted:\n        //   * if the insertion index is greater than the head (regardless of circular array form)\n        //      -> shift the preceding elements\n        //   * otherwise, the circular array has (2) form and the insertion index is less than tail\n        //      -> shift all elements in the back of the array\n        //      -> shift preceding elements in the front of the array\n        // In case the succeeding elements are shifted:\n        //   * if the insertion index is less than the tail (regardless of circular array form)\n        //      -> shift the succeeding elements\n        //   * otherwise, the circular array has (2) form and the insertion index is greater than head\n        //      -> shift all elements in the front of the array\n        //      -> shift succeeding elements in the back of the array\n\n        val internalIndex = internalIndex(index)\n\n        if (index < (size + 1) shr 1) {\n            // closer to the first element -> shift preceding elements\n            val decrementedInternalIndex = decremented(internalIndex)\n            val decrementedHead = decremented(head)\n\n            if (decrementedInternalIndex >= head) {\n                elementData[decrementedHead] = elementData[head]  // head can be zero\n                elementData.copyInto(elementData, head, head + 1, decrementedInternalIndex + 1)\n            } else { // head > tail\n                elementData.copyInto(elementData, head - 1, head, elementData.size) // head can't be zero\n                elementData[elementData.size - 1] = elementData[0]\n                elementData.copyInto(elementData, 0, 1, decrementedInternalIndex + 1)\n            }\n\n            elementData[decrementedInternalIndex] = element\n            head = decrementedHead\n        } else {\n            // closer to the last element -> shift succeeding elements\n            val tail = internalIndex(size)\n\n            if (internalIndex < tail) {\n                elementData.copyInto(elementData, internalIndex + 1, internalIndex, tail)\n            } else { // head > tail\n                elementData.copyInto(elementData, 1, 0, tail)\n                elementData[0] = elementData[elementData.size - 1]\n                elementData.copyInto(elementData, internalIndex + 1, internalIndex, elementData.size - 1)\n            }\n\n            elementData[internalIndex] = element\n        }\n        size += 1\n    }\n\n    private fun copyCollectionElements(internalIndex: Int, elements: Collection<E>) {\n        val iterator = elements.iterator()\n\n        for (index in internalIndex until elementData.size) {\n            if (!iterator.hasNext()) break\n            elementData[index] = iterator.next()\n        }\n        for (index in 0 until head) {\n            if (!iterator.hasNext()) break\n            elementData[index] = iterator.next()\n        }\n\n        size += elements.size\n    }\n\n    public override fun addAll(elements: Collection<E>): Boolean {\n        if (elements.isEmpty()) return false\n        ensureCapacity(this.size + elements.size)\n        copyCollectionElements(internalIndex(size), elements)\n        return true\n    }\n\n    public override fun addAll(index: Int, elements: Collection<E>): Boolean {\n        AbstractList.checkPositionIndex(index, size)\n\n        if (elements.isEmpty()) {\n            return false\n        } else if (index == size) {\n            return addAll(elements)\n        }\n\n        ensureCapacity(this.size + elements.size)\n\n        val tail = internalIndex(size)\n        val internalIndex = internalIndex(index)\n        val elementsSize = elements.size\n\n        if (index < (size + 1) shr 1) {\n            // closer to the first element -> shift preceding elements\n\n            var shiftedHead = head - elementsSize\n\n            if (internalIndex >= head) {\n                if (shiftedHead >= 0) {\n                    elementData.copyInto(elementData, shiftedHead, head, internalIndex)\n                } else { // head < tail, insertion leads to head >= tail\n                    shiftedHead += elementData.size\n                    val elementsToShift = internalIndex - head\n                    val shiftToBack = elementData.size - shiftedHead\n\n                    if (shiftToBack >= elementsToShift) {\n                        elementData.copyInto(elementData, shiftedHead, head, internalIndex)\n                    } else {\n                        elementData.copyInto(elementData, shiftedHead, head, head + shiftToBack)\n                        elementData.copyInto(elementData, 0, head + shiftToBack, internalIndex)\n                    }\n                }\n            } else { // head > tail, internalIndex < tail\n                elementData.copyInto(elementData, shiftedHead, head, elementData.size)\n                if (elementsSize >= internalIndex) {\n                    elementData.copyInto(elementData, elementData.size - elementsSize, 0, internalIndex)\n                } else {\n                    elementData.copyInto(elementData, elementData.size - elementsSize, 0, elementsSize)\n                    elementData.copyInto(elementData, 0, elementsSize, internalIndex)\n                }\n            }\n            head = shiftedHead\n            copyCollectionElements(negativeMod(internalIndex - elementsSize), elements)\n        } else {\n            // closer to the last element -> shift succeeding elements\n\n            val shiftedInternalIndex = internalIndex + elementsSize\n\n            if (internalIndex < tail) {\n                if (tail + elementsSize <=
```

```
elementData.size) {\n                elementData.copyInto(elementData, shiftedInternalIndex, internalIndex, tail)\n
    } else { // head < tail, insertion leads to head >= tail\n                if (shiftedInternalIndex >= elementData.size)
{\n                elementData.copyInto(elementData, shiftedInternalIndex - elementData.size, internalIndex, tail)\n
            } else {\n                val shiftToFront = tail + elementsSize - elementData.size\n
elementData.copyInto(elementData, 0, tail - shiftToFront, tail)\n                elementData.copyInto(elementData,
shiftedInternalIndex, internalIndex, tail - shiftToFront)\n            }\n        }\n    } else { // head > tail,
internalIndex > head\n            elementData.copyInto(elementData, elementsSize, 0, tail)\n            if
(shiftedInternalIndex >= elementData.size) {\n                elementData.copyInto(elementData, shiftedInternalIndex
- elementData.size, internalIndex, elementData.size)\n            } else {\n
elementData.copyInto(elementData, 0, elementData.size - elementsSize, elementData.size)\n
elementData.copyInto(elementData, shiftedInternalIndex, internalIndex, elementData.size - elementsSize)\n
}\n        }\n        copyCollectionElements(internalIndex, elements)\n    }\n\n    return true\n  }\n\n  public
override fun get(index: Int): E {\n    AbstractList.checkElementIndex(index, size)\n\n    return
internalGet(internalIndex(index))\n  }\n\n  public override fun set(index: Int, element: E): E {\n
AbstractList.checkElementIndex(index, size)\n\n    val internalIndex = internalIndex(index)\n    val oldElement
= internalGet(internalIndex)\n    elementData[internalIndex] = element\n\n    return oldElement\n  }\n\n
public override fun contains(element: E): Boolean = indexOf(element) != -1\n\n  public override fun
indexOf(element: E): Int {\n    val tail = internalIndex(size)\n\n    if (head < tail) {\n      for (index in head
until tail) {\n        if (element == elementData[index]) return index - head\n      }\n    } else if (head >=
tail) {\n      for (index in head until elementData.size) {\n        if (element == elementData[index]) return
index - head\n      }\n      for (index in 0 until tail) {\n        if (element == elementData[index]) return
index + elementData.size - head\n      }\n    }\n\n    return -1\n  }\n\n  public override fun
lastIndexOf(element: E): Int {\n    val tail = internalIndex(size)\n\n    if (head < tail) {\n      for (index in tail
- 1 downTo head) {\n        if (element == elementData[index]) return index - head\n      }\n    } else if
(head > tail) {\n      for (index in tail - 1 downTo 0) {\n        if (element == elementData[index]) return
index + elementData.size - head\n      }\n      for (index in elementData.lastIndex downTo head) {\n
if (element == elementData[index]) return index - head\n      }\n    }\n\n    return -1\n  }\n\n  public
override fun remove(element: E): Boolean {\n    val index = indexOf(element)\n    if (index == -1) return
false\n    removeAt(index)\n    return true\n  }\n\n  public override fun removeAt(index: Int): E {\n
AbstractList.checkElementIndex(index, size)\n\n    if (index == lastIndex) {\n      return removeLast()\n    }
else if (index == 0) {\n      return removeFirst()\n    }\n\n    val internalIndex = internalIndex(index)\n
val element = internalGet(internalIndex)\n\n    if (index < size shr 1) {\n      // closer to the first element ->
shift preceding elements\n      if (internalIndex >= head) {\n        elementData.copyInto(elementData, head
+ 1, head, internalIndex)\n      } else { // head > tail, internalIndex < head\n
elementData.copyInto(elementData, 1, 0, internalIndex)\n        elementData[0] = elementData[elementData.size
- 1]\n        elementData.copyInto(elementData, head + 1, head, elementData.size - 1)\n      }\n\n
elementData[head] = null\n      head = incremented(head)\n    } else {\n      // closer to the last element ->
shift succeeding elements\n      val internalLastIndex = internalIndex(lastIndex)\n\n      if (internalIndex <=
internalLastIndex) {\n        elementData.copyInto(elementData, internalIndex, internalIndex + 1,
internalLastIndex + 1)\n      } else { // head > tail, internalIndex > head\n
elementData.copyInto(elementData, internalIndex, internalIndex + 1, elementData.size)\n
elementData[elementData.size - 1] = elementData[0]\n        elementData.copyInto(elementData, 0, 1,
internalLastIndex + 1)\n      }\n\n      elementData[internalLastIndex] = null\n    }\n    size -= 1\n\n
return element\n  }\n\n  public override fun removeAll(elements: Collection<E>): Boolean = filterInPlace {
!elements.contains(it) }\n\n  public override fun retainAll(elements: Collection<E>): Boolean = filterInPlace {
elements.contains(it) }\n\n  private inline fun filterInPlace(predicate: (E) -> Boolean): Boolean {\n    if
(this.isEmpty() || elementData.isEmpty())\n      return false\n\n    val tail = internalIndex(size)\n    var
newTail = head\n    var modified = false\n\n    if (head < tail) {\n      for (index in head until tail) {\n
```

```
 val element = elementData[index]\n\n                @Suppress(\"UNCHECKED_CAST\")\n                if (predicate(element as E))\n                    elementData[newTail++] = element\n                else\n                    modified = true\n            }\n\n            elementData.fill(null, newTail, tail)\n\n        } else {\n            for (index in head until elementData.size) {\n                val element = elementData[index]\n                elementData[index] = null\n\n                @Suppress(\"UNCHECKED_CAST\")\n                if (predicate(element as E))\n                    elementData[newTail++] = element\n                else\n                    modified = true\n            }\n\n            newTail = positiveMod(newTail)\n\n            for (index in 0 until tail) {\n                val element = elementData[index]\n                elementData[index] = null\n\n                @Suppress(\"UNCHECKED_CAST\")\n                if (predicate(element as E)) {\n                    elementData[newTail] = element\n                    newTail = incremented(newTail)\n                } else {\n                    modified = true\n                }\n            }\n        }\n        if (modified)\n            size = negativeMod(newTail - head)\n\n        return modified\n    }\n\n    public override fun clear() {\n        val tail = internalIndex(size)\n        if (head < tail) {\n            elementData.fill(null, head, tail)\n        } else if (isNotEmpty()) {\n            elementData.fill(null, head, elementData.size)\n            elementData.fill(null, 0, tail)\n        }\n        head = 0\n        size = 0\n    }\n\n    @Suppress(\"NOTHING_TO_OVERRIDE\")\n    override fun <T> toArray(array: Array<T>): Array<T> {\n        @Suppress(\"UNCHECKED_CAST\")\n        val dest = (if (array.size >= size) array else arrayOfNulls(array, size)) as Array<Any?>\n\n        val tail = internalIndex(size)\n        if (head < tail) {\n            elementData.copyInto(dest, startIndex = head, endIndex = tail)\n        } else if (isNotEmpty()) {\n            elementData.copyInto(dest, destinationOffset = 0, startIndex = head, endIndex = elementData.size)\n            elementData.copyInto(dest, destinationOffset = elementData.size - head, startIndex = 0, endIndex = tail)\n        }\n        if (dest.size > size) {\n            dest[size] = null // null-terminate\n        }\n\n        @Suppress(\"UNCHECKED_CAST\")\n        return dest as Array<T>\n    }\n\n    @Suppress(\"NOTHING_TO_OVERRIDE\")\n    override fun toArray(): Array<Any?> {\n        return toArray(arrayOfNulls<Any?>(size))\n    }\n\n    // for testing\n    internal fun <T> testToArray(array: Array<T>): Array<T> = toArray(array)\n    internal fun testToArray(): Array<Any?> = toArray()\n\n    internal companion object {\n        private val emptyElementData = emptyArray<Any?>()\n        private const val maxArraySize = Int.MAX_VALUE - 8\n        private const val defaultMinCapacity = 10\n\n        internal fun newCapacity(oldCapacity: Int, minCapacity: Int): Int {\n            // overflow-conscious\n            var newCapacity = oldCapacity + (oldCapacity shr 1)\n            if (newCapacity - minCapacity < 0)\n                newCapacity = minCapacity\n            if (newCapacity - maxArraySize > 0)\n                newCapacity = if (minCapacity > maxArraySize) Int.MAX_VALUE else maxArraySize\n            return newCapacity\n        }\n    }\n\n    // For testing only\n    internal fun internalStructure(structure: (head: Int, elements: Array<Any?>) -> Unit) {\n        val tail = internalIndex(size)\n        val head = if (isEmpty() || head < tail) head else head - elementData.size\n        structure(head, toArray())\n    }\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"ArraysKt\")\n\n\npackage kotlin.collections\n\nimport kotlin.contracts.*\n\n\n/**\n * Returns a single list of all elements from all arrays in the given array.\n * @sample samples.collections.Arrays.Transformations.flattenArray\n */\npublic fun <T> Array<out Array<out T>>.flatten(): List<T> {\n    val result = ArrayList<T>(sumOf { it.size })\n    for (element in this) {\n        result.addAll(element)\n    }\n    return result\n}\n\n/**\n * Returns a pair of lists, where\n * *first* list is built from the first values of each pair from this array,\n * *second* list is built from the second values of each pair from this array.\n * @sample samples.collections.Arrays.Transformations.unzipArray\n */\npublic fun <T, R> Array<out Pair<T, R>>.unzip(): Pair<List<T>, List<R>> {\n    val listT = ArrayList<T>(size)\n    val listR = ArrayList<R>(size)\n    for (pair in this) {\n        listT.add(pair.first)\n        listR.add(pair.second)\n    }\n    return listT to listR\n}\n\n/**\n * Returns `true` if this nullable array is either null or empty.\n * @sample samples.collections.Arrays.Usage.arrayIsNullOrEmpty\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun Array<*>?.isNullOrEmpty(): Boolean {\n    contract {\n        returns(false) implies (this@isNullOrEmpty != null)\n    }\n\n    return this == null ||
```

```
this.isEmpty()\n}\n\n/**\n * Returns this array if it's not empty\n * or the result of calling [defaultValue] function if the array is empty.\n *\n * @sample samples.collections.Arrays.Usage.arrayIfEmpty\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Suppress(\"UPPER_BOUND_CANNOT_BE_ARRAY\")\npublic inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : Array<*>, C : R =\n    if (isEmpty()) defaultValue() else this\n\n\n@OptIn(ExperimentalUnsignedTypes::class)\n@SinceKotlin(\"1.3\")\n@PublishedApi\n@kotlin.jvm.JvmName(\"contentDeepEquals\")\n@kotlin.js.JsName(\"contentDeepEqualsImpl\")\ninternal fun <T> Array<out T>?.contentDeepEqualsImpl(other: Array<out T>?): Boolean {\n    if (this === other) return true\n    if (this == null || other == null || this.size != other.size) return false\n\n    for (i in indices) {\n        val v1 = this[i]\n        val v2 = other[i]\n\n        if (v1 === v2) {\n            continue\n        } else if (v1 == null || v2 == null) {\n            return false\n        }\n\n        when {\n            v1 is Array<*>     && v2 is Array<*>     -> if (!v1.contentDeepEquals(v2)) return false\n            v1 is ByteArray    && v2 is ByteArray    -> if (!v1.contentEquals(v2)) return false\n            v1 is ShortArray   && v2 is ShortArray   -> if (!v1.contentEquals(v2)) return false\n            v1 is IntArray     && v2 is IntArray     -> if (!v1.contentEquals(v2)) return false\n            v1 is LongArray    && v2 is LongArray    -> if (!v1.contentEquals(v2)) return false\n            v1 is FloatArray   && v2 is FloatArray   -> if (!v1.contentEquals(v2)) return false\n            v1 is DoubleArray  && v2 is DoubleArray  -> if (!v1.contentEquals(v2)) return false\n            v1 is CharArray    && v2 is CharArray    -> if (!v1.contentEquals(v2)) return false\n            v1 is BooleanArray && v2 is BooleanArray -> if (!v1.contentEquals(v2)) return false\n\n            v1 is UByteArray   && v2 is UByteArray   -> if (!v1.contentEquals(v2)) return false\n            v1 is UShortArray  && v2 is UShortArray  -> if (!v1.contentEquals(v2)) return false\n            v1 is UIntArray    && v2 is UIntArray    -> if (!v1.contentEquals(v2)) return false\n            v1 is ULongArray   && v2 is ULongArray   -> if (!v1.contentEquals(v2)) return false\n\n            else -> if (v1 != v2) return false\n        }\n    }\n    return true\n}\n\n@SinceKotlin(\"1.3\")\n@PublishedApi\n@kotlin.jvm.JvmName(\"contentDeepToString\")\n@kotlin.js.JsName(\"contentDeepToStringImpl\")\ninternal fun <T> Array<out T>?.contentDeepToStringImpl(): String {\n    if (this == null) return \"null\"\n    val length = size.coerceAtMost((Int.MAX_VALUE - 2) / 5) * 5 + 2 // in order not to overflow Int.MAX_VALUE\n    return buildString(length) {\n        contentDeepToStringInternal(this, mutableListOf())\n    }\n}\n\n@OptIn(ExperimentalUnsignedTypes::class)\nprivate fun <T> Array<out T>.contentDeepToStringInternal(result: StringBuilder, processed: MutableList<Array<*>>) {\n    if (this in processed) {\n        result.append(\"[...]\")\n        return\n    }\n    processed.add(this)\n    result.append('[')\n\n    for (i in indices) {\n        if (i != 0) {\n            result.append(\", \")\n        }\n        val element = this[i]\n        when (element) {\n            null            -> result.append(\"null\")\n            is Array<*>     -> element.contentDeepToStringInternal(result, processed)\n            is ByteArray    -> result.append(element.contentToString())\n            is ShortArray   -> result.append(element.contentToString())\n            is IntArray     -> result.append(element.contentToString())\n            is LongArray    -> result.append(element.contentToString())\n            is FloatArray   -> result.append(element.contentToString())\n            is DoubleArray  -> result.append(element.contentToString())\n            is CharArray    -> result.append(element.contentToString())\n            is BooleanArray -> result.append(element.contentToString())\n\n            is UByteArray   -> result.append(element.contentToString())\n            is UShortArray  -> result.append(element.contentToString())\n            is UIntArray    -> result.append(element.contentToString())\n            is ULongArray   -> result.append(element.contentToString())\n\n            else            -> result.append(element.toString())\n        }\n    }\n\n    result.append(']')\n    processed.removeAt(processed.lastIndex)\n}","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n\n/** Returns true if the brittle contains optimization is enabled. See KT-45438. */\ninternal expect fun brittleContainsOptimizationEnabled(): Boolean\n\n/**\n * Returns true if [brittleContainsOptimizationEnabled] is true\n * and it's safe to convert this collection to a set without changing contains method behavior.\n */\nprivate fun <T> Collection<T>.safeToConvertToSet() = brittleContainsOptimizationEnabled() && size > 2 && this is ArrayList\n\n/**\n * When
```

[brittleContainsOptimizationEnabled] is true:\n * - Converts this [Iterable] to a set if it is not a [Collection].\n * - Converts this [Collection] to a set, when it's worth so and it doesn't change contains method behavior.\n * - Otherwise returns this.\n * When [brittleContainsOptimizationEnabled] is false:\n * - Converts this [Iterable] to a list if it is not a [Collection].\n * - Otherwise returns this.\n */\ninternal fun <T> Iterable<T>.convertToSetForSetOperationWith(source: Iterable<T>): Collection<T> =\n when (this) {\n is Set -> this\n is Collection ->\n when {\n source is Collection && source.size < 2 -> this\n else -> if (this.safeToConvertToSet()) toHashSet() else this\n }\n else -> if (brittleContainsOptimizationEnabled()) toHashSet() else toList()\n }\n\n/**\n * When [brittleContainsOptimizationEnabled] is true:\n * - Converts this [Iterable] to a set if it is not a [Collection].\n * - Converts this [Collection] to a set, when it's worth so and it doesn't change contains method behavior.\n * - Otherwise returns this.\n * When [brittleContainsOptimizationEnabled] is false:\n * - Converts this [Iterable] to a list if it is not a [Collection].\n * - Otherwise returns this.\n */\ninternal fun <T> Iterable<T>.convertToSetForSetOperation(): Collection<T> =\n when (this) {\n is Set -> this\n is Collection -> if (this.safeToConvertToSet()) toHashSet() else this\n else -> if (brittleContainsOptimizationEnabled()) toHashSet() else toList()\n }\n\n/**\n * Converts this sequence to a set if [brittleContainsOptimizationEnabled] is true,\n * otherwise converts it to a list.\n */\ninternal fun <T> Sequence<T>.convertToSetForSetOperation(): Collection<T> =\n if (brittleContainsOptimizationEnabled()) toHashSet() else toList()\n\n/**\n * Converts this array to a set if [brittleContainsOptimizationEnabled] is true,\n * otherwise converts it to a list.\n */\ninternal fun <T> Array<T>.convertToSetForSetOperation(): Collection<T> =\n if (brittleContainsOptimizationEnabled()) toHashSet() else asList()","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n/**\n * Data class representing a value from a collection or sequence, along with its index in that collection or sequence.\n *\n * @property value the underlying value.\n * @property index the index of the value in the collection or sequence.\n */\npublic data class IndexedValue<out T>(public val index: Int, public val value: T)\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmName(\"MapAccessorsKt\")\n\npackage kotlin.collections\n\nimport kotlin.reflect.KProperty\nimport kotlin.internal.Exact\n\n/**\n * Returns the value of the property for the given object from this read-only map.\n * @param thisRef the object for which the value is requested (not used).\n * @param property the metadata for the property, used to get the name of property and lookup the value corresponding to this name in the map.\n * @return the property value.\n *\n * @throws NoSuchElementException when the map doesn't contain value for the property name and doesn't provide an implicit default (see [withDefault]).\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <V, V1 : V> Map<in String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =\n @Suppress(\"UNCHECKED_CAST\") (getOrImplicitDefault(property.name) as V1)\n\n/**\n * Returns the value of the property for the given object from this mutable map.\n * @param thisRef the object for which the value is requested (not used).\n * @param property the metadata for the property, used to get the name of property and lookup the value corresponding to this name in the map.\n * @return the property value.\n *\n * @throws NoSuchElementException when the map doesn't contain value for the property name and doesn't provide an implicit default (see [withDefault]).\n */\n@kotlin.jvm.JvmName(\"getVar\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <V, V1 : V> MutableMap<in String, out @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =\n @Suppress(\"UNCHECKED_CAST\") (getOrImplicitDefault(property.name) as V1)\n\n/**\n * Stores the value of the property for the given object in this mutable map.\n * @param thisRef the object for which the value is requested (not used).\n * @param property the metadata for the property, used to get the name of property and store the value associated with that name in the map.\n * @param value the value to set.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <V> MutableMap<in String, in V>.setValue(thisRef: Any?, property: KProperty<*>, value: V) {\n this.put(property.name, value)\n}\n","/*\n * Copyright 2010-2018

JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"MapsKt\")\n\npackage kotlin.collections\n\n/**\n * Returns the value for the given key, or the implicit default value for this map.\n * By default no implicit value is provided for maps and a [NoSuchElementException] is thrown.\n * To create a map with implicit default value use [withDefault] method.\n *\n * @throws NoSuchElementException when the map doesn't contain a value for the specified key and no implicit default was provided for that map.\n */\n@kotlin.jvm.JvmName(\"getOrImplicitDefaultNullable\")\n@PublishedApi\ninternal fun <K, V> Map<K, V>.getOrImplicitDefault(key: K): V {\n   if (this is MapWithDefault)\n   return this.getOrImplicitDefault(key)\n\n   return getOrElseNullable(key, { throw NoSuchElementException(\"Key $key is missing in the map.\") })\n}\n\n/**\n * Returns a wrapper of this read-only map, having the implicit default value provided with the specified function [defaultValue].\n *\n * This implicit default value is used when the original map doesn't contain a value for the key specified\n * and a value is obtained with [Map.getValue] function, for example when properties are delegated to the map.\n *\n * When this map already has an implicit default value provided with a former call to [withDefault], it is being replaced by this call.\n */\npublic fun <K, V> Map<K, V>.withDefault(defaultValue: (key: K) -> V): Map<K, V> =\n   when (this) {\n   is MapWithDefault -> this.map.withDefault(defaultValue)\n   else -> MapWithDefaultImpl(this, defaultValue)\n   }\n\n/**\n * Returns a wrapper of this mutable map, having the implicit default value provided with the specified function [defaultValue].\n *\n * This implicit default value is used when the original map doesn't contain a value for the key specified\n * and a value is obtained with [Map.getValue] function, for example when properties are delegated to the map.\n *\n * When this map already has an implicit default value provided with a former call to [withDefault], it is being replaced by this call.\n */\n@kotlin.jvm.JvmName(\"withDefaultMutable\")\npublic fun <K, V> MutableMap<K, V>.withDefault(defaultValue: (key: K) -> V): MutableMap<K, V> =\n   when (this) {\n   is MutableMapWithDefault -> this.map.withDefault(defaultValue)\n   else -> MutableMapWithDefaultImpl(this, defaultValue)\n   }\n\n\nprivate interface MapWithDefault<K, out V> : Map<K, V> {\n   public val map: Map<K, V>\n   public fun getOrImplicitDefault(key: K): V\n}\n\nprivate interface MutableMapWithDefault<K, V> : MutableMap<K, V>, MapWithDefault<K, V> {\n   public override val map: MutableMap<K, V>\n}\n\n\nprivate class MapWithDefaultImpl<K, out V>(public override val map: Map<K, V>, private val default: (key: K) -> V) : MapWithDefault<K, V> {\n   override fun equals(other: Any?): Boolean = map.equals(other)\n   override fun hashCode(): Int = map.hashCode()\n   override fun toString(): String = map.toString()\n   override val size: Int get() = map.size\n   override fun isEmpty(): Boolean = map.isEmpty()\n   override fun containsKey(key: K): Boolean = map.containsKey(key)\n   override fun containsValue(value: @UnsafeVariance V): Boolean = map.containsValue(value)\n   override fun get(key: K): V? = map.get(key)\n   override val keys: Set<K> get() = map.keys\n   override val values: Collection<V> get() = map.values\n   override val entries: Set<Map.Entry<K, V>> get() = map.entries\n\n   override fun getOrImplicitDefault(key: K): V = map.getOrElseNullable(key, { default(key) })\n}\n\nprivate class MutableMapWithDefaultImpl<K, V>(public override val map: MutableMap<K, V>, private val default: (key: K) -> V) : MutableMapWithDefault<K, V> {\n   override fun equals(other: Any?): Boolean = map.equals(other)\n   override fun hashCode(): Int = map.hashCode()\n   override fun toString(): String = map.toString()\n   override val size: Int get() = map.size\n   override fun isEmpty(): Boolean = map.isEmpty()\n   override fun containsKey(key: K): Boolean = map.containsKey(key)\n   override fun containsValue(value: @UnsafeVariance V): Boolean = map.containsValue(value)\n   override fun get(key: K): V? = map.get(key)\n   override val keys: MutableSet<K> get() = map.keys\n   override val values: MutableCollection<V> get() = map.values\n   override val entries: MutableSet<MutableMap.MutableEntry<K, V>> get() = map.entries\n\n   override fun put(key: K, value: V): V? = map.put(key, value)\n   override fun remove(key: K): V? = map.remove(key)\n   override fun putAll(from: Map<out K, V>) = map.putAll(from)\n   override fun clear() = map.clear()\n\n   override fun getOrImplicitDefault(key: K): V = map.getOrElseNullable(key, { default(key) })\n}\n\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n

*/\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"CollectionsKt\")\n\npackage kotlin.collections\n\nimport kotlin.random.Random\n\n/**\n * Removes a single instance of the specified element from this\n * collection, if it is present.\n *\n * Allows to overcome type-safety restriction of `remove` that requires to pass an element of type `E`.\n *\n * @return `true` if the element has been successfully removed; `false` if it was not present in the collection.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <@kotlin.internal.OnlyInputTypes T> MutableCollection<out T>.remove(element: T): Boolean =\n   @Suppress(\"UNCHECKED_CAST\") (this as MutableCollection<T>).remove(element)\n\n/**\n * Removes all of this collection's elements that are also contained in the specified collection.\n\n * Allows to overcome type-safety restriction of `removeAll` that requires to pass a collection of type `Collection<E>`.\n *\n * @return `true` if any of the specified elements was removed from the collection, `false` if the collection was not modified.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <@kotlin.internal.OnlyInputTypes T> MutableCollection<out T>.removeAll(elements: Collection<T>): Boolean =\n   @Suppress(\"UNCHECKED_CAST\") (this as MutableCollection<T>).removeAll(elements)\n\n/**\n * Retains only the elements in this collection that are contained in the specified collection.\n *\n * Allows to overcome type-safety restriction of `retainAll` that requires to pass a collection of type `Collection<E>`.\n *\n * @return `true` if any element was removed from the collection, `false` if the collection was not modified.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <@kotlin.internal.OnlyInputTypes T> MutableCollection<out T>.retainAll(elements: Collection<T>): Boolean =\n   @Suppress(\"UNCHECKED_CAST\") (this as MutableCollection<T>).retainAll(elements)\n\n/**\n * Adds the specified [element] to this mutable collection.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> MutableCollection<in T>.plusAssign(element: T) {\n   this.add(element)\n}\n\n/**\n * Adds all elements of the given [elements] collection to this mutable collection.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Iterable<T>) {\n   this.addAll(elements)\n}\n\n/**\n * Adds all elements of the given [elements] array to this mutable collection.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Array<T>) {\n   this.addAll(elements)\n}\n\n/**\n * Adds all elements of the given [elements] sequence to this mutable collection.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Sequence<T>) {\n   this.addAll(elements)\n}\n\n/**\n * Removes a single instance of the specified [element] from this mutable collection.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> MutableCollection<in T>.minusAssign(element: T) {\n   this.remove(element)\n}\n\n/**\n * Removes all elements contained in the given [elements] collection from this mutable collection.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Iterable<T>) {\n   this.removeAll(elements)\n}\n\n/**\n * Removes all elements contained in the given [elements] array from this mutable collection.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Array<T>) {\n   this.removeAll(elements)\n}\n\n/**\n * Removes all elements contained in the given [elements] sequence from this mutable collection.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Sequence<T>) {\n   this.removeAll(elements)\n}\n\n/**\n * Adds all elements of the given [elements] collection to this [MutableCollection].\n */\npublic fun <T> MutableCollection<in T>.addAll(elements: Iterable<T>): Boolean {\n   when (elements) {\n   is Collection -> return addAll(elements)\n   else -> {\n   var result: Boolean = false\n   for (item in elements)\n   if (add(item)) result = true\n   return result\n   }\n   }\n}\n\n/**\n * Adds all elements of the given [elements] sequence to this [MutableCollection].\n */\npublic fun <T> MutableCollection<in T>.addAll(elements: Sequence<T>): Boolean {\n   var result: Boolean = false\n   for (item in elements) {\n   if (add(item)) result = true\n   }\n   return result\n}\n\n/**\n * Adds all elements of the given [elements] array to this [MutableCollection].\n */\npublic fun <T> MutableCollection<in T>.addAll(elements: Array<out T>): Boolean {\n   return addAll(elements.asList())\n}\n\n/**\n * Removes all elements from this [MutableCollection] that are also contained in the given [elements] collection.\n */\npublic fun <T> MutableCollection<in T>.removeAll(elements: Iterable<T>): Boolean {\n   return removeAll(elements.convertToSetForSetOperationWith(this))\n}\n\n/**\n * Removes all elements from this [MutableCollection] that are also contained in the given [elements] sequence.\n

*/\npublic fun <T> MutableCollection<in T>.removeAll(elements: Sequence<T>): Boolean {\n   val set = elements.convertToSetForSetOperation()\n   return set.isNotEmpty() && removeAll(set)\n}\n\n/**\n * Removes all elements from this [MutableCollection] that are also contained in the given [elements] array.\n */\npublic fun <T> MutableCollection<in T>.removeAll(elements: Array<out T>): Boolean {\n   return elements.isNotEmpty() && removeAll(elements.convertToSetForSetOperation())\n}\n\n/**\n * Retains only elements of this [MutableCollection] that are contained in the given [elements] collection.\n */\npublic fun <T> MutableCollection<in T>.retainAll(elements: Iterable<T>): Boolean {\n   return retainAll(elements.convertToSetForSetOperationWith(this))\n}\n\n/**\n * Retains only elements of this [MutableCollection] that are contained in the given [elements] array.\n */\npublic fun <T> MutableCollection<in T>.retainAll(elements: Array<out T>): Boolean {\n   if (elements.isNotEmpty())\n   return retainAll(elements.convertToSetForSetOperation())\n   else\n   return retainNothing()\n}\n\n/**\n * Retains only elements of this [MutableCollection] that are contained in the given [elements] sequence.\n */\npublic fun <T> MutableCollection<in T>.retainAll(elements: Sequence<T>): Boolean {\n   val set = elements.convertToSetForSetOperation()\n   if (set.isNotEmpty())\n   return retainAll(set)\n   else\n   return retainNothing()\n}\n\nprivate fun MutableCollection<*>.retainNothing(): Boolean {\n   val result = isNotEmpty()\n   clear()\n   return result\n}\n\n\n/**\n * Removes all elements from this [MutableIterable] that match the given [predicate].\n *\n * @return `true` if any element was removed from this collection, or `false` when no elements were removed and collection was not modified.\n */\npublic fun <T> MutableIterable<T>.removeAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, true)\n\n/**\n * Retains only elements of this [MutableIterable] that match the given [predicate].\n *\n * @return `true` if any element was removed from this collection, or `false` when all elements were retained and collection was not modified.\n */\npublic fun <T> MutableIterable<T>.retainAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, false)\n\nprivate fun <T> MutableIterable<T>.filterInPlace(predicate: (T) -> Boolean, predicateResultToRemove: Boolean): Boolean {\n   var result = false\n   with(iterator()) {\n   while (hasNext())\n   if (predicate(next()) == predicateResultToRemove) {\n   remove()\n   result = true\n   }\n   }\n   return result\n}\n\n\n/**\n * Removes the element at the specified [index] from this list.\n * In Kotlin one should use the [MutableList.removeAt] function instead.\n */\n@Deprecated(\"Use removeAt(index) instead.\", ReplaceWith(\"removeAt(index)\"), level = DeprecationLevel.ERROR)\n@kotlin.internal.InlineOnly\npublic inline fun <T> MutableList<T>.remove(index: Int): T = removeAt(index)\n\n/**\n * Removes the first element from this mutable list and returns that removed element, or throws [NoSuchElementException] if this list is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun <T> MutableList<T>.removeFirst(): T = if (isEmpty()) throw NoSuchElementException(\"List is empty.\") else removeAt(0)\n\n/**\n * Removes the first element from this mutable list and returns that removed element, or returns `null` if this list is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun <T> MutableList<T>.removeFirstOrNull(): T? = if (isEmpty()) null else removeAt(0)\n\n/**\n * Removes the last element from this mutable list and returns that removed element, or throws [NoSuchElementException] if this list is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun <T> MutableList<T>.removeLast(): T = if (isEmpty()) throw NoSuchElementException(\"List is empty.\") else removeAt(lastIndex)\n\n/**\n * Removes the last element from this mutable list and returns that removed element, or returns `null` if this list is empty.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun <T> MutableList<T>.removeLastOrNull(): T? = if (isEmpty()) null else removeAt(lastIndex)\n\n/**\n * Removes all elements from this [MutableList] that match the given [predicate].\n *\n * @return `true` if any element was removed from this collection, or `false` when no elements were removed and collection was not modified.\n */\npublic fun <T> MutableList<T>.removeAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, true)\n\n/**\n * Retains only elements of this [MutableList] that match the given [predicate].\n *\n * @return `true` if any element was removed from this collection, or `false` when all elements were retained and collection was not

modified.\n */\npublic fun <T> MutableList<T>.retainAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, false)\n\nprivate fun <T> MutableList<T>.filterInPlace(predicate: (T) -> Boolean, predicateResultToRemove: Boolean): Boolean {\n   if (this !is RandomAccess)\n   return (this as MutableIterable<T>).filterInPlace(predicate, predicateResultToRemove)\n\n   var writeIndex: Int = 0\n   for (readIndex in 0..lastIndex) {\n   val element = this[readIndex]\n   if (predicate(element) == predicateResultToRemove)\n   continue\n\n   if (writeIndex != readIndex)\n   this[writeIndex] = element\n\n   writeIndex++\n   }\n   if (writeIndex < size) {\n   for (removeIndex in lastIndex downTo writeIndex)\n   removeAt(removeIndex)\n\n   return true\n   } else {\n   return false\n   }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"CollectionsKt\")\n\npackage kotlin.collections\n\nprivate open class ReversedListReadOnly<out T>(private val delegate: List<T>) : AbstractList<T>() {\n   override val size: Int get() = delegate.size\n   override fun get(index: Int): T = delegate[reverseElementIndex(index)]\n}\n\nprivate class ReversedList<T>(private val delegate: MutableList<T>) : AbstractMutableList<T>() {\n   override val size: Int get() = delegate.size\n   override fun get(index: Int): T = delegate[reverseElementIndex(index)]\n\n   override fun clear() = delegate.clear()\n   override fun removeAt(index: Int): T = delegate.removeAt(reverseElementIndex(index))\n\n   override fun set(index: Int, element: T): T = delegate.set(reverseElementIndex(index), element)\n   override fun add(index: Int, element: T) {\n   delegate.add(reversePositionIndex(index), element)\n   }\n}\n\nprivate fun List<*>.reverseElementIndex(index: Int) =\n   if (index in 0..lastIndex) lastIndex - index else throw IndexOutOfBoundsException(\"Element index $index must be in range [${0..lastIndex}].\")\n\nprivate fun List<*>.reversePositionIndex(index: Int) =\n   if (index in 0..size) size - index else throw IndexOutOfBoundsException(\"Position index $index must be in range [${0..size}].\")\n\n\n/**\n * Returns a reversed read-only view of the original List.\n * All changes made in the original list will be reflected in the reversed one.\n * @sample samples.collections.ReversedViews.asReversedList\n */\npublic fun <T> List<T>.asReversed(): List<T> = ReversedListReadOnly(this)\n\n/**\n * Returns a reversed mutable view of the original mutable List.\n * All changes made in the original list will be reflected in the reversed one and vice versa.\n * @sample samples.collections.ReversedViews.asReversedMutableList\n */\n@kotlin.jvm.JvmName(\"asReversedMutable\")\npublic fun <T> MutableList<T>.asReversed(): MutableList<T> = ReversedList(this)\n\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"SequencesKt\")\n@file:OptIn(ExperimentalTypeInference::class)\n\npackage kotlin.sequences\n\nimport kotlin.coroutines.*\nimport kotlin.coroutines.intrinsics.*\nimport kotlin.experimental.ExperimentalTypeInference\n\n/**\n * Builds a [Sequence] lazily yielding values one by one.\n *\n * @see kotlin.sequences.generateSequence\n *\n * @sample samples.collections.Sequences.Building.buildSequenceYieldAll\n * @sample samples.collections.Sequences.Building.buildFibonacciSequence\n */\n@SinceKotlin(\"1.3\")\npublic fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence { iterator(block) }\n\n@SinceKotlin(\"1.3\")\n@Deprecated(\"Use 'sequence { }' function instead.\", ReplaceWith(\"sequence(builderAction)\"), level = DeprecationLevel.ERROR)\n@kotlin.internal.InlineOnly\npublic inline fun <T> buildSequence(@BuilderInference noinline builderAction: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence { iterator(builderAction) }\n\n/**\n * Builds an [Iterator] lazily yielding values one by one.\n *\n * @sample samples.collections.Sequences.Building.buildIterator\n * @sample samples.collections.Iterables.Building.iterable\n */\n@SinceKotlin(\"1.3\")\npublic fun <T> iterator(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Iterator<T> {\n   val iterator = SequenceBuilderIterator<T>()\n   iterator.nextStep = block.createCoroutineUnintercepted(receiver = iterator, completion = iterator)\n   return iterator\n}\n\n@SinceKotlin(\"1.3\")\n@Deprecated(\"Use 'iterator { }' function instead.\",

ReplaceWith(\"iterator(builderAction)\"), level = DeprecationLevel.ERROR)\n@kotlin.internal.InlineOnly\npublic inline fun <T> buildIterator(@BuilderInference noinline builderAction: suspend SequenceScope<T>.() -> Unit): Iterator<T> = iterator(builderAction)\n\n/**\n * The scope for yielding values of a [Sequence] or an [Iterator], provides [yield] and [yieldAll] suspension functions.\n *\n * @see sequence\n * @see iterator\n *\n * @sample samples.collections.Sequences.Building.buildSequenceYieldAll\n * @sample samples.collections.Sequences.Building.buildFibonacciSequence\n */\n@RestrictsSuspension\n@SinceKotlin(\"1.3\")\npublic abstract class SequenceScope<in T> internal constructor() {\n /**\n * Yields a value to the [Iterator] being built and suspends\n * until the next value is requested.\n *\n * @sample samples.collections.Sequences.Building.buildSequenceYieldAll\n * @sample samples.collections.Sequences.Building.buildFibonacciSequence\n */\n public abstract suspend fun yield(value: T)\n\n /**\n * Yields all values from the `iterator` to the [Iterator] being built\n * and suspends until all these values are iterated and the next one is requested.\n *\n * The sequence of values returned by the given iterator can be potentially infinite.\n *\n * @sample samples.collections.Sequences.Building.buildSequenceYieldAll\n */\n public abstract suspend fun yieldAll(iterator: Iterator<T>)\n\n /**\n * Yields a collections of values to the [Iterator] being built\n * and suspends until all these values are iterated and the next one is requested.\n *\n * @sample samples.collections.Sequences.Building.buildSequenceYieldAll\n */\n public suspend fun yieldAll(elements: Iterable<T>) {\n if (elements is Collection && elements.isEmpty()) return\n return yieldAll(elements.iterator())\n }\n\n /**\n * Yields potentially infinite sequence of values to the [Iterator] being built\n * and suspends until all these values are iterated and the next one is requested.\n *\n * The sequence can be potentially infinite.\n *\n * @sample samples.collections.Sequences.Building.buildSequenceYieldAll\n */\n public suspend fun yieldAll(sequence: Sequence<T>) = yieldAll(sequence.iterator())\n}\n\n@Deprecated(\"Use SequenceScope class instead.\", ReplaceWith(\"SequenceScope<T>\"), level = DeprecationLevel.ERROR)\npublic typealias SequenceBuilder<T> = SequenceScope<T>\n\nprivate typealias State = Int\n\nprivate const val State_NotReady: State = 0\nprivate const val State_ManyNotReady: State = 1\nprivate const val State_ManyReady: State = 2\nprivate const val State_Ready: State = 3\nprivate const val State_Done: State = 4\nprivate const val State_Failed: State = 5\n\nprivate class SequenceBuilderIterator<T> : SequenceScope<T>(), Iterator<T>, Continuation<Unit> {\n private var state = State_NotReady\n private var nextValue: T? = null\n private var nextIterator: Iterator<T>? = null\n var nextStep: Continuation<Unit>? = null\n\n override fun hasNext(): Boolean {\n while (true) {\n when (state) {\n State_NotReady -> {}\n State_ManyNotReady ->\n if (nextIterator!!.hasNext()) {\n state = State_ManyReady\n return true\n } else {\n nextIterator = null\n }\n State_Done -> return false\n State_Ready, State_ManyReady -> return true\n else -> throw exceptionalState()\n }\n\n state = State_Failed\n val step = nextStep!!\n nextStep = null\n step.resume(Unit)\n }\n }\n\n override fun next(): T {\n when (state) {\n State_NotReady, State_ManyNotReady -> return nextNotReady()\n State_ManyReady -> {\n state = State_ManyNotReady\n return nextIterator!!.next()\n }\n State_Ready -> {\n state = State_NotReady\n @Suppress(\"UNCHECKED_CAST\")\n val result = nextValue as T\n nextValue = null\n return result\n }\n else -> throw exceptionalState()\n }\n }\n\n private fun nextNotReady(): T {\n if (!hasNext()) throw NoSuchElementException() else return next()\n }\n\n private fun exceptionalState(): Throwable = when (state) {\n State_Done -> NoSuchElementException()\n State_Failed -> IllegalStateException(\"Iterator has failed.\")\n else -> IllegalStateException(\"Unexpected state of the iterator: $state\")\n }\n\n\n override suspend fun yield(value: T) {\n nextValue = value\n state = State_Ready\n return suspendCoroutineUninterceptedOrReturn { c ->\n nextStep = c\n COROUTINE_SUSPENDED\n }\n }\n\n override suspend fun yieldAll(iterator: Iterator<T>) {\n if (!iterator.hasNext()) return\n nextIterator = iterator\n state = State_ManyReady\n return suspendCoroutineUninterceptedOrReturn { c ->\n nextStep = c\n COROUTINE_SUSPENDED\n }\n }\n\n // Completion continuation implementation\n override fun resumeWith(result: Result<Unit>) {\n

result.getOrThrow() // just rethrow exception if it is there\n state = State_Done\n }\n\n override val context: CoroutineContext\n get() = EmptyCoroutineContext\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\ninternal fun checkWindowSizeStep(size: Int, step: Int) {\n require(size > 0 && step > 0) {\n if (size != step)\n \"Both size $size and step $step must be greater than zero.\"\n else\n \"size $size must be greater than zero.\"\n }\n}\n\ninternal fun <T> Sequence<T>.windowedSequence(size: Int, step: Int, partialWindows: Boolean, reuseBuffer: Boolean): Sequence<List<T>> {\n checkWindowSizeStep(size, step)\n return Sequence { windowedIterator(iterator(), size, step, partialWindows, reuseBuffer) }\n}\n\ninternal fun <T> windowedIterator(iterator: Iterator<T>, size: Int, step: Int, partialWindows: Boolean, reuseBuffer: Boolean): Iterator<List<T>> {\n if (!iterator.hasNext()) return EmptyIterator\n return iterator<List<T>> {\n val bufferInitialCapacity = size.coerceAtMost(1024)\n val gap = step - size\n if (gap >= 0) {\n var buffer = ArrayList<T>(bufferInitialCapacity)\n var skip = 0\n for (e in iterator) {\n if (skip > 0) { skip -= 1; continue }\n buffer.add(e)\n if (buffer.size == size) {\n yield(buffer)\n if (reuseBuffer) buffer.clear() else buffer = ArrayList(size)\n skip = gap\n }\n }\n if (buffer.isNotEmpty()) {\n if (partialWindows || buffer.size == size) yield(buffer)\n }\n } else {\n var buffer = RingBuffer<T>(bufferInitialCapacity)\n for (e in iterator) {\n buffer.add(e)\n if (buffer.isFull()) {\n if (buffer.size < size) { buffer = buffer.expanded(maxCapacity = size); continue }\n\n yield(if (reuseBuffer) buffer else ArrayList(buffer))\n buffer.removeFirst(step)\n }\n }\n if (partialWindows) {\n while (buffer.size > step) {\n yield(if (reuseBuffer) buffer else ArrayList(buffer))\n buffer.removeFirst(step)\n }\n if (buffer.isNotEmpty()) yield(buffer)\n }\n }\n}\n\ninternal class MovingSubList<out E>(private val list: List<E>) : AbstractList<E>(), RandomAccess {\n private var fromIndex: Int = 0\n private var _size: Int = 0\n\n fun move(fromIndex: Int, toIndex: Int) {\n checkRangeIndexes(fromIndex, toIndex, list.size)\n this.fromIndex = fromIndex\n this._size = toIndex - fromIndex\n }\n\n override fun get(index: Int): E {\n checkElementIndex(index, _size)\n\n return list[fromIndex + index]\n }\n\n override val size: Int get() = _size\n}\n\n\n/**\n * Provides ring buffer implementation.\n *\n * Buffer overflow is not allowed so [add] doesn't overwrite tail but raises an exception.\n */\nprivate class RingBuffer<T>(private val buffer: Array<Any?>, filledSize: Int) : AbstractList<T>(), RandomAccess {\n init {\n require(filledSize >= 0) { \"ring buffer filled size should not be negative but it is $filledSize\" }\n require(filledSize <= buffer.size) { \"ring buffer filled size: $filledSize cannot be larger than the buffer size: ${buffer.size}\" }\n }\n\n constructor(capacity: Int) : this(arrayOfNulls<Any?>(capacity), 0)\n\n private val capacity = buffer.size\n private var startIndex: Int = 0\n\n override var size: Int = filledSize\n private set\n\n override fun get(index: Int): T {\n checkElementIndex(index, size)\n @Suppress(\"UNCHECKED_CAST\")\n return buffer[startIndex.forward(index)] as T\n }\n\n fun isFull() = size == capacity\n\n override fun iterator(): Iterator<T> = object : AbstractIterator<T>() {\n private var count = size\n private var index = startIndex\n\n override fun computeNext() {\n if (count == 0) {\n done()\n } else {\n @Suppress(\"UNCHECKED_CAST\")\n setNext(buffer[index] as T)\n index = index.forward(1)\n count--\n }\n }\n }\n\n @Suppress(\"UNCHECKED_CAST\")\n override fun <T> toArray(array: Array<T>): Array<T> {\n val result: Array<T?> =\n if (array.size < this.size) array.copyOf(this.size) else array as Array<T?>\n\n val size = this.size\n\n var widx = 0\n var idx = startIndex\n\n while (widx < size && idx < capacity) {\n result[widx] = buffer[idx] as T\n widx++\n idx++\n }\n\n idx = 0\n while (widx < size) {\n result[widx] = buffer[idx] as T\n widx++\n idx++\n }\n if (result.size > this.size) result[this.size] = null\n\n return result as Array<T>\n }\n\n override fun toArray(): Array<Any?> {\n return toArray(arrayOfNulls(size))\n }\n\n /**\n * Creates a new ring buffer with the capacity equal to the minimum of [maxCapacity] and 1.5 * [capacity].\n * The returned ring buffer contains the same elements as this ring buffer.\n */\n fun expanded(maxCapacity: Int): RingBuffer<T> {\n val newCapacity = (capacity +

(capacity shr 1) + 1).coerceAtMost(maxCapacity)\n val newBuffer = if (startIndex == 0) buffer.copyOf(newCapacity) else toArray(arrayOfNulls(newCapacity))\n return RingBuffer(newBuffer, size)\n }\n\n /**\n * Add [element] to the buffer or fail with [IllegalStateException] if no free space available in the buffer\n */\n fun add(element: T) {\n if (isFull()) {\n throw IllegalStateException(\"ring buffer is full\")\n }\n\n buffer[startIndex.forward(size)] = element\n size++\n }\n\n /**\n * Removes [n] first elements from the buffer or fails with [IllegalArgumentException] if not enough elements in the buffer to remove\n */\n fun removeFirst(n: Int) {\n require(n >= 0) { \"n shouldn't be negative but it is $n\" }\n require(n <= size) { \"n shouldn't be greater than the buffer size: n = $n, size = $size\" }\n\n if (n > 0) {\n val start = startIndex\n val end = start.forward(n)\n\n if (start > end) {\n buffer.fill(null, start, capacity)\n buffer.fill(null, 0, end)\n } else {\n buffer.fill(null, start, end)\n }\n\n startIndex = end\n size -= n\n }\n }\n\n @Suppress(\"NOTHING_TO_INLINE\")\n private inline fun Int.forward(n: Int): Int = (this + n) % capacity\n}\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n// UByteArray ==\n@ExperimentalUnsignedTypes\nprivate fun partition(\n array: UByteArray, left: Int, right: Int): Int {\n var i = left\n var j = right\n val pivot = array[(left + right) / 2]\n while (i <= j) {\n while (array[i] < pivot)\n i++\n while (array[j] > pivot)\n j--\n if (i <= j) {\n val tmp = array[i]\n array[i] = array[j]\n array[j] = tmp\n i++\n j--\n }\n }\n return i\n}\n\n@ExperimentalUnsignedTypes\nprivate fun quickSort(\n array: UByteArray, left: Int, right: Int) {\n val index = partition(array, left, right)\n if (left < index - 1)\n quickSort(array, left, index - 1)\n if (index < right)\n quickSort(array, index, right)\n}\n\n// UShortArray ==\n@ExperimentalUnsignedTypes\nprivate fun partition(\n array: UShortArray, left: Int, right: Int): Int {\n var i = left\n var j = right\n val pivot = array[(left + right) / 2]\n while (i <= j) {\n while (array[i] < pivot)\n i++\n while (array[j] > pivot)\n j--\n if (i <= j) {\n val tmp = array[i]\n array[i] = array[j]\n array[j] = tmp\n i++\n j--\n }\n }\n return i\n}\n\n@ExperimentalUnsignedTypes\nprivate fun quickSort(\n array: UShortArray, left: Int, right: Int) {\n val index = partition(array, left, right)\n if (left < index - 1)\n quickSort(array, left, index - 1)\n if (index < right)\n quickSort(array, index, right)\n}\n\n// UIntArray ==\n@ExperimentalUnsignedTypes\nprivate fun partition(\n array: UIntArray, left: Int, right: Int): Int {\n var i = left\n var j = right\n val pivot = array[(left + right) / 2]\n while (i <= j) {\n while (array[i] < pivot)\n i++\n while (array[j] > pivot)\n j--\n if (i <= j) {\n val tmp = array[i]\n array[i] = array[j]\n array[j] = tmp\n i++\n j--\n }\n }\n return i\n}\n\n@ExperimentalUnsignedTypes\nprivate fun quickSort(\n array: UIntArray, left: Int, right: Int) {\n val index = partition(array, left, right)\n if (left < index - 1)\n quickSort(array, left, index - 1)\n if (index < right)\n quickSort(array, index, right)\n}\n\n// ULongArray ==\n@ExperimentalUnsignedTypes\nprivate fun partition(\n array: ULongArray, left: Int, right: Int): Int {\n var i = left\n var j = right\n val pivot = array[(left + right) / 2]\n while (i <= j) {\n while (array[i] < pivot)\n i++\n while (array[j] > pivot)\n j--\n if (i <= j) {\n val tmp = array[i]\n array[i] = array[j]\n array[j] = tmp\n i++\n j--\n }\n }\n return i\n}\n\n@ExperimentalUnsignedTypes\nprivate fun quickSort(\n array: ULongArray, left: Int, right: Int) {\n val index = partition(array, left, right)\n if (left < index - 1)\n quickSort(array, left, index - 1)\n if (index < right)\n quickSort(array, index, right)\n}\n\n// Interfaces ==\n/**\n * Sorts the given array using qsort algorithm.\n */\n@ExperimentalUnsignedTypes\ninternal fun sortArray(array:

UByteArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)\n@ExperimentalUnsignedTypes\ninternal fun sortArray(array: UShortArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)\n@ExperimentalUnsignedTypes\ninternal fun sortArray(array: UIntArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)\n@ExperimentalUnsignedTypes\ninternal fun sortArray(array: ULongArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\nimport kotlin.internal.InlineOnly\n\n/**\n * Compares this object with the specified object for order. Returns zero if this object is equal\n * to the specified [other] object, a negative number if it's less than [other], or a positive number\n * if it's greater than [other].\n *\n * This function delegates to [Comparable.compareTo] and allows to call it in infix form.\n */\n@InlineOnly\n@SinceKotlin(\"1.6\")\npublic inline infix fun <T> Comparable<T>.compareTo(other: T): Int =\n this.compareTo(other)\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.contracts\n\nimport kotlin.internal.ContractsDsl\nimport kotlin.internal.InlineOnly\n\n/**\n * This marker distinguishes the experimental contract declaration API and is used to opt-in for that feature\n * when declaring contracts of user functions.\n *\n * Any usage of a declaration annotated with `@ExperimentalContracts` must be accepted either by\n * annotating that usage with the [OptIn] annotation, e.g. `@OptIn(ExperimentalContracts::class)`,\n * or by using the compiler argument `-opt-in=kotlin.contracts.ExperimentalContracts`.\n */\n@Retention(AnnotationRetention.BINARY)\n@SinceKotlin(\"1.3\")\n@RequiresOptIn\n@MustBeDocumented\npublic annotation class ExperimentalContracts\n\n/**\n * Provides a scope, where the functions of the contract DSL, such as [returns], [callsInPlace], etc.,\n * can be used to describe the contract of a function.\n *\n * This type is used as a receiver type of the lambda function passed to the [contract] function.\n *\n * @see contract\n */\n@ContractsDsl\n@ExperimentalContracts\n@SinceKotlin(\"1.3\")\npublic interface ContractBuilder {\n /**\n   * Describes a situation when a function returns normally, without any exceptions thrown.\n   *\n   * Use [SimpleEffect.implies] function to describe a conditional effect that happens in such case.\n   *\n   */\n // @sample samples.contracts.returnsContract\n @ContractsDsl public fun returns(): Returns\n\n /**\n   * Describes a situation when a function returns normally with the specified return [value].\n   *\n   * The possible values of [value] are limited to `true`, `false` or `null`.\n   *\n   * Use [SimpleEffect.implies] function to describe a conditional effect that happens in such case.\n   *\n   */\n // @sample samples.contracts.returnsTrueContract\n // @sample samples.contracts.returnsFalseContract\n // @sample samples.contracts.returnsNullContract\n @ContractsDsl public fun returns(value: Any?): Returns\n\n /**\n   * Describes a situation when a function returns normally with any value that is not `null`.\n   *\n   * Use [SimpleEffect.implies] function to describe a conditional effect that happens in such case.\n   *\n   */\n // @sample samples.contracts.returnsNotNullContract\n @ContractsDsl public fun returnsNotNull(): ReturnsNotNull\n\n /**\n   * Specifies that the function parameter [lambda] is invoked in place.\n   *\n   * This contract specifies that:\n   * 1. the function [lambda] can only be invoked during the call of the owner function,\n   * and it won't be invoked after that owner function call is completed;\n   * 2. _(optionally)_ the function [lambda] is invoked the amount of times specified by the [kind] parameter,\n   * see the [InvocationKind] enum for possible values.\n   *\n   * A function declaring the `callsInPlace` effect must be _inline_.\n   *\n   */\n /* @sample samples.contracts.callsInPlaceAtMostOnceContract\n   * @sample samples.contracts.callsInPlaceAtLeastOnceContract\n   * @sample samples.contracts.callsInPlaceExactlyOnceContract\n   * @sample samples.contracts.callsInPlaceUnknownContract\n   */\n @ContractsDsl public fun <R> callsInPlace(lambda: Function<R>, kind: InvocationKind = InvocationKind.UNKNOWN): CallsInPlace\n}\n\n/**\n * Specifies how many times a function invokes its function parameter in place.\n *\n * See [ContractBuilder.callsInPlace] for the details of the call-in-place function contract.\n

*/\n@ContractsDsl\n@ExperimentalContracts\n@SinceKotlin(\"1.3\")\npublic enum class InvocationKind {\n /**\n * A function parameter will be invoked one time or not invoked at all.\n */\n // @sample samples.contracts.callsInPlaceAtMostOnceContract\n @ContractsDsl AT_MOST_ONCE,\n\n /**\n * A function parameter will be invoked one or more times.\n *\n */\n // @sample samples.contracts.callsInPlaceAtLeastOnceContract\n @ContractsDsl AT_LEAST_ONCE,\n\n /**\n * A function parameter will be invoked exactly one time.\n *\n */\n // @sample samples.contracts.callsInPlaceExactlyOnceContract\n @ContractsDsl EXACTLY_ONCE,\n\n /**\n * A function parameter is called in place, but it's unknown how many times it can be called.\n *\n */\n // @sample samples.contracts.callsInPlaceUnknownContract\n @ContractsDsl UNKNOWN\n}\n\n/**\n * Specifies the contract of a function.\n *\n * The contract description must be at the beginning of a function and have at least one effect.\n *\n * Only the top-level functions can have a contract for now.\n *\n * @param builder the lambda where the contract of a function is described with the help of the [ContractBuilder] members.\n *\n */\n/* @sample samples.contracts.returnsContract\n* @sample samples.contracts.returnsTrueContract\n* @sample samples.contracts.returnsFalseContract\n* @sample samples.contracts.returnsNullContract\n* @sample samples.contracts.returnsNotNullContract\n* @sample samples.contracts.callsInPlaceAtMostOnceContract\n* @sample samples.contracts.callsInPlaceAtLeastOnceContract\n* @sample samples.contracts.callsInPlaceExactlyOnceContract\n* @sample samples.contracts.callsInPlaceUnknownContract\n*/\n@ContractsDsl\n@ExperimentalContracts\n@InlineOnly\n@SinceKotlin(\"1.3\")\n@Suppress(\"UNUSED_PARAMETER\")\npublic inline fun contract(builder: ContractBuilder.() -> Unit) { }\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.coroutines\n\n/**\n * Marks coroutine context element that intercepts coroutine continuations.\n * The coroutines framework uses [ContinuationInterceptor.Key] to retrieve the interceptor and\n * intercepts all coroutine continuations with [interceptContinuation] invocations.\n *\n * [ContinuationInterceptor] behaves like a [polymorphic element][AbstractCoroutineContextKey], meaning that\n * its implementation delegates [get][CoroutineContext.Element.get] and [minusKey][CoroutineContext.Element.minusKey]\n * to [getPolymorphicElement] and [minusPolymorphicKey] respectively.\n * [ContinuationInterceptor] subtypes can be extracted from the coroutine context using either [ContinuationInterceptor.Key]\n * or subtype key if it extends [AbstractCoroutineContextKey].\n */\n@SinceKotlin(\"1.3\")\npublic interface ContinuationInterceptor : CoroutineContext.Element {\n /**\n * The key that defines *the* context interceptor.\n */\n companion object Key : CoroutineContext.Key<ContinuationInterceptor>\n\n /**\n * Returns continuation that wraps the original [continuation], thus intercepting all resumptions.\n * This function is invoked by coroutines framework when needed and the resulting continuations are\n * cached internally per each instance of the original [continuation].\n *\n * This function may simply return original [continuation] if it does not want to intercept this particular continuation.\n *\n * When the original [continuation] completes, coroutine framework invokes [releaseInterceptedContinuation]\n * with the resulting continuation if it was intercepted, that is if `interceptContinuation` had previously\n * returned a different continuation instance.\n */\n public fun <T> interceptContinuation(continuation: Continuation<T>): Continuation<T>\n\n /**\n * Invoked for the continuation instance returned by [interceptContinuation] when the original\n * continuation completes and will not be used anymore. This function is invoked only if [interceptContinuation]\n * had returned a different continuation instance from the one it was invoked with.\n *\n * Default implementation does nothing.\n *\n * @param continuation Continuation instance returned by this interceptor's [interceptContinuation] invocation.\n */\n public fun releaseInterceptedContinuation(continuation: Continuation<*>) {\n /* do nothing by default */\n }\n\n public override operator fun <E : CoroutineContext.Element> get(key: CoroutineContext.Key<E>): E? {\n // getPolymorphicKey specialized for ContinuationInterceptor key\n @OptIn(ExperimentalStdlibApi::class)\n if (key is AbstractCoroutineContextKey<*, *>) {\n @Suppress(\"UNCHECKED_CAST\")\n return if (key.isSubKey(this.key)) key.tryCast(this) as? E else

null\n }\n @Suppress(\"UNCHECKED_CAST\")\n return if (ContinuationInterceptor === key) this as E else null\n }\n\n\n public override fun minusKey(key: CoroutineContext.Key<*>): CoroutineContext {\n // minusPolymorphicKey specialized for ContinuationInterceptor key\n @OptIn(ExperimentalStdlibApi::class)\n if (key is AbstractCoroutineContextKey<*, *>) {\n return if (key.isSubKey(this.key) && key.tryCast(this) != null) EmptyCoroutineContext else this\n }\n return if (ContinuationInterceptor === key) EmptyCoroutineContext else this\n }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.coroutines\n\n/**\n * Persistent context for the coroutine. It is an indexed set of [Element] instances.\n * An indexed set is a mix between a set and a map.\n * Every element in this set has a unique [Key].\n */\n@SinceKotlin(\"1.3\")\npublic interface CoroutineContext {\n /**\n * Returns the element with the given [key] from this context or `null`.\n */\n public operator fun <E : Element> get(key: Key<E>): E?\n\n /**\n * Accumulates entries of this context starting with [initial] value and applying [operation]\n * from left to right to current accumulator value and each element of this context.\n */\n public fun <R> fold(initial: R, operation: (R, Element) -> R): R\n\n /**\n * Returns a context containing elements from this context and elements from other [context].\n * The elements from this context with the same key as in the other one are dropped.\n */\n public operator fun plus(context: CoroutineContext): CoroutineContext =\n if (context === EmptyCoroutineContext) this else // fast path -- avoid lambda creation\n context.fold(this) { acc, element ->\n val removed = acc.minusKey(element.key)\n if (removed === EmptyCoroutineContext) element else {\n // make sure interceptor is always last in the context (and thus is fast to get when present)\n val interceptor = removed[ContinuationInterceptor]\n if (interceptor == null) CombinedContext(removed, element) else {\n val left = removed.minusKey(ContinuationInterceptor)\n if (left === EmptyCoroutineContext) CombinedContext(element, interceptor) else\n CombinedContext(CombinedContext(left, element), interceptor)\n }\n }\n }\n\n /**\n * Returns a context containing elements from this context, but without an element with\n * the specified [key].\n */\n public fun minusKey(key: Key<*>): CoroutineContext\n\n /**\n * Key for the elements of [CoroutineContext]. [E] is a type of element with this key.\n */\n public interface Key<E : Element>\n\n /**\n * An element of the [CoroutineContext]. An element of the coroutine context is a singleton context by itself.\n */\n public interface Element : CoroutineContext {\n /**\n * A key of this coroutine context element.\n */\n public val key: Key<*>\n\n public override operator fun <E : Element> get(key: Key<E>): E? =\n @Suppress(\"UNCHECKED_CAST\")\n if (this.key == key) this as E else null\n\n public override fun <R> fold(initial: R, operation: (R, Element) -> R): R =\n operation(initial, this)\n\n public override fun minusKey(key: Key<*>): CoroutineContext =\n if (this.key == key) EmptyCoroutineContext else this\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.coroutines\n\nimport kotlin.coroutines.CoroutineContext.Element\nimport kotlin.coroutines.CoroutineContext.Key\n\n/**\n * Base class for [CoroutineContext.Element] implementations.\n */\n@SinceKotlin(\"1.3\")\npublic abstract class AbstractCoroutineContextElement(public override val key: Key<*>) : Element\n\n/**\n * Base class for [CoroutineContext.Key] associated with polymorphic [CoroutineContext.Element] implementation.\n * Polymorphic element implementation implies delegating its [get][Element.get] and [minusKey][Element.minusKey]\n * to [getPolymorphicElement] and [minusPolymorphicKey] respectively.\n *\n * Polymorphic elements can be extracted from the coroutine context using both element key and its supertype key.\n * Example of polymorphic elements:\n * ```\n * open class BaseElement : CoroutineContext.Element {\n * companion object Key : CoroutineContext.Key<BaseElement>\n * override val key: CoroutineContext.Key<*> get() = Key\n * // It is important to use getPolymorphicKey and minusPolymorphicKey\n * override fun <E : CoroutineContext.Element> get(key: CoroutineContext.Key<E>): E? = getPolymorphicElement(key)\n * override fun minusKey(key: CoroutineContext.Key<*>): CoroutineContext = minusPolymorphicKey(key)\n * }\n *\n * class DerivedElement : BaseElement() {\n *

```
companion object Key : AbstractCoroutineContextKey<BaseElement, DerivedElement>(BaseElement, { it as? DerivedElement })\n * }\n * // Now it is possible to query both `BaseElement` and `DerivedElement`\n * someContext[BaseElement] // Returns BaseElement?, non-null both for BaseElement and DerivedElement instances\n * someContext[DerivedElement] // Returns DerivedElement?, non-null only for DerivedElement instance\n * ```\n * @param B base class of a polymorphic element\n * @param baseKey an instance of base key\n * @param E element type associated with the current key\n * @param safeCast a function that can safely cast abstract [CoroutineContext.Element] to the concrete [E] type\n *                 and return the element if it is a subtype of [E] or `null` otherwise.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalStdlibApi\npublic abstract class AbstractCoroutineContextKey<B : Element, E : B>(\n    baseKey: Key<B>,\n    private val safeCast: (element: Element) -> E?\n) : Key<E> {\n    private val topmostKey: Key<*> = if (baseKey is AbstractCoroutineContextKey<*, *>) baseKey.topmostKey else baseKey\n\n    internal fun tryCast(element: Element): E? = safeCast(element)\n    internal fun isSubKey(key: Key<*>): Boolean = key === this || topmostKey === key\n}\n\n/**\n * Returns the current element if it is associated with the given [key] in a polymorphic manner or `null` otherwise.\n * This method returns non-null value if either [Element.key] is equal to the given [key] or if the [key] is associated\n * with [Element.key] via [AbstractCoroutineContextKey].\n * See [AbstractCoroutineContextKey] for the example of usage.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalStdlibApi\npublic fun <E : Element> Element.getPolymorphicElement(key: Key<E>): E? {\n    if (key is AbstractCoroutineContextKey<*, *>) {\n        @Suppress(\"UNCHECKED_CAST\")\n        return if (key.isSubKey(this.key)) key.tryCast(this) as? E else null\n    }\n    @Suppress(\"UNCHECKED_CAST\")\n    return if (this.key === key) this as E else null\n}\n\n/**\n * Returns empty coroutine context if the element is associated with the given [key] in a polymorphic manner\n * or `null` otherwise.\n * This method returns empty context if either [Element.key] is equal to the given [key] or if the [key] is associated\n * with [Element.key] via [AbstractCoroutineContextKey].\n * See [AbstractCoroutineContextKey] for the example of usage.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalStdlibApi\npublic fun Element.minusPolymorphicKey(key: Key<*>): CoroutineContext {\n    if (key is AbstractCoroutineContextKey<*, *>) {\n        return if (key.isSubKey(this.key) && key.tryCast(this) != null) EmptyCoroutineContext else this\n    }\n    return if (this.key === key) EmptyCoroutineContext else this\n}\n\n/**\n * An empty coroutine context.\n */\n@SinceKotlin(\"1.3\")\npublic object EmptyCoroutineContext : CoroutineContext, Serializable {\n    private const val serialVersionUID: Long = 0\n    private fun readResolve(): Any = EmptyCoroutineContext\n\n    public override fun <E : Element> get(key: Key<E>): E? = null\n    public override fun <R> fold(initial: R, operation: (R, Element) -> R): R = initial\n    public override fun plus(context: CoroutineContext): CoroutineContext = context\n    public override fun minusKey(key: Key<*>): CoroutineContext = this\n    public override fun hashCode(): Int = 0\n    public override fun toString(): String = \"EmptyCoroutineContext\"\n}\n\n//---------------------- internal impl ----------------------\n\n// this class is not exposed, but is hidden inside implementations\n// this is a left-biased list, so that `plus` works naturally\n@SinceKotlin(\"1.3\")\ninternal class CombinedContext(\n    private val left: CoroutineContext,\n    private val element: Element\n) : CoroutineContext, Serializable {\n\n    override fun <E : Element> get(key: Key<E>): E? {\n        var cur = this\n        while (true) {\n            cur.element[key]?.let { return it }\n            val next = cur.left\n            if (next is CombinedContext) {\n                cur = next\n            } else {\n                return next[key]\n            }\n        }\n    }\n\n    public override fun <R> fold(initial: R, operation: (R, Element) -> R): R =\n        operation(left.fold(initial, operation), element)\n\n    public override fun minusKey(key: Key<*>): CoroutineContext {\n        element[key]?.let { return left }\n        val newLeft = left.minusKey(key)\n        return when {\n            newLeft === left -> this\n            newLeft === EmptyCoroutineContext -> element\n            else -> CombinedContext(newLeft, element)\n        }\n    }\n\n    private fun size(): Int {\n        var cur = this\n        var size = 2\n        while (true) {\n            cur = cur.left as? CombinedContext ?: return size\n            size++\n        }\n    }\n\n    private fun contains(element: Element): Boolean =\n        get(element.key) == element\n\n    private fun containsAll(context: CombinedContext): Boolean {\n        var cur = context\n        while (true) {\n            if (!contains(cur.element)) return false\n            val next = cur.left\n            if (next is CombinedContext) {\n
```

cur = next\n } else {\n return contains(next as Element)\n }\n }\n }\n\n override fun equals(other: Any?): Boolean =\n this === other || other is CombinedContext && other.size() == size() && other.containsAll(this)\n\n override fun hashCode(): Int = left.hashCode() + element.hashCode()\n\n override fun toString(): String =\n \"[\" + fold(\"\") { acc, element ->\n if (acc.isEmpty()) element.toString() else \"$acc, $element\"\n } + \"]\"\n\n private fun writeReplace(): Any {\n val n = size()\n val elements = arrayOfNulls<CoroutineContext>(n)\n var index = 0\n fold(Unit) { _, element -> elements[index++] = element }\n check(index == n)\n @Suppress(\"UNCHECKED_CAST\")\n return Serialized(elements as Array<CoroutineContext>)\n }\n\n private class Serialized(val elements: Array<CoroutineContext>) : Serializable {\n companion object {\n private const val serialVersionUID: Long = 0L\n }\n\n private fun readResolve(): Any = elements.fold(EmptyCoroutineContext, CoroutineContext::plus)\n }\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmName(\"IntrinsicsKt\")\n@file:kotlin.jvm.JvmMultifileClass\n\npackage kotlin.coroutines.intrinsics\n\nimport kotlin.contracts.*\nimport kotlin.coroutines.*\nimport kotlin.internal.InlineOnly\n\n/**\n * Obtains the current continuation instance inside suspend functions and either suspends\n * currently running coroutine or returns result immediately without suspension.\n *\n * If the [block] returns the special [COROUTINE_SUSPENDED] value, it means that suspend function did suspend the execution and will\n * not return any result immediately. In this case, the [Continuation] provided to the [block] shall be\n * resumed by invoking [Continuation.resumeWith] at some moment in the\n * future when the result becomes available to resume the computation.\n *\n * Otherwise, the return value of the [block] must have a type assignable to [T] and represents the result of this suspend function.\n * It means that the execution was not suspended and the [Continuation] provided to the [block] shall not be invoked.\n * As the result type of the [block] is declared as `Any?` and cannot be correctly type-checked,\n * its proper return type remains on the conscience of the suspend function's author.\n *\n * Invocation of [Continuation.resumeWith] resumes coroutine directly in the invoker's thread without going through the\n * [ContinuationInterceptor] that might be present in the coroutine's [CoroutineContext].\n * It is the invoker's responsibility to ensure that a proper invocation context is established.\n * [Continuation.intercepted] can be used to acquire the intercepted continuation.\n *\n * Note that it is not recommended to call either [Continuation.resume] nor [Continuation.resumeWithException] functions synchronously\n * in the same stackframe where suspension function is run. Use [suspendCoroutine] as a safer way to obtain current\n * continuation instance.\n */\n@SinceKotlin(\"1.3\")\n@InlineOnly\n@Suppress(\"UNUSED_PARAMETER\", \"RedundantSuspendModifier\")\npublic suspend inline fun <T> suspendCoroutineUninterceptedOrReturn(crossinline block: (Continuation<T>) -> Any?): T {\n contract { callsInPlace(block, InvocationKind.EXACTLY_ONCE) }\n throw NotImplementedError(\"Implementation of suspendCoroutineUninterceptedOrReturn is intrinsic\")\n}\n\n/**\n * This value is used as a return value of [suspendCoroutineUninterceptedOrReturn] `block` argument to state that\n * the execution was suspended and will not return any result immediately.\n *\n * **Note: this value should not be used in general code.** Using it outside of the context of\n * `suspendCoroutineUninterceptedOrReturn` function return value (including, but not limited to,\n * storing this value in other properties, returning it from other functions, etc)\n * can lead to unspecified behavior of the code.\n */\n// It is implemented as property with getter to avoid ProGuard <clinit> problem with multifile IntrinsicsKt class\n@SinceKotlin(\"1.3\")\npublic val COROUTINE_SUSPENDED: Any get() = CoroutineSingletons.COROUTINE_SUSPENDED\n\n// Using enum here ensures two important properties:\n// 1. It makes SafeContinuation serializable with all kinds of serialization frameworks (since all of them natively support enums)\n// 2. It improves debugging experience, since you clearly see toString() value of those objects and what package they come from\n@SinceKotlin(\"1.3\")\n@PublishedApi // This class is Published API via serialized representation of SafeContinuation, don't rename/move\ninternal enum class CoroutineSingletons { COROUTINE_SUSPENDED, UNDECIDED, RESUMED }\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that

can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.experimental\n\n/** Performs a bitwise AND operation between the two values. */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline infix fun Byte.and(other: Byte): Byte = (this.toInt() and other.toInt()).toByte()\n\n/** Performs a bitwise OR operation between the two values. */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline infix fun Byte.or(other: Byte): Byte = (this.toInt() or other.toInt()).toByte()\n\n/** Performs a bitwise XOR operation between the two values. */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline infix fun Byte.xor(other: Byte): Byte = (this.toInt() xor other.toInt()).toByte()\n\n/** Inverts the bits in this value. */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun Byte.inv(): Byte = (this.toInt().inv()).toByte()\n\n\n/** Performs a bitwise AND operation between the two values. */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline infix fun Short.and(other: Short): Short = (this.toInt() and other.toInt()).toShort()\n\n/** Performs a bitwise OR operation between the two values. */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline infix fun Short.or(other: Short): Short = (this.toInt() or other.toInt()).toShort()\n\n/** Performs a bitwise XOR operation between the two values. */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline infix fun Short.xor(other: Short): Short = (this.toInt() xor other.toInt()).toShort()\n\n/** Inverts the bits in this value. */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic inline fun Short.inv(): Short = (this.toInt().inv()).toShort()\n\n\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.experimental\n\n/**\n * The experimental marker for type inference augmenting annotations.\n *\n * Any usage of a declaration annotated with `@ExperimentalTypeInference` must be accepted either by\n * annotating that usage with the [OptIn] annotation, e.g. `@OptIn(ExperimentalTypeInference::class)`,\n * or by using the compiler argument `-opt-in=kotlin.experimental.ExperimentalTypeInference`.\n */\n@RequiresOptIn(level = RequiresOptIn.Level.ERROR)\n@MustBeDocumented\n@Retention(AnnotationRetention.BINARY)\n@Target(AnnotationTarget.ANNOTATION_CLASS)\n@SinceKotlin(\"1.3\")\npublic annotation class ExperimentalTypeInference\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.internal\n\n/**\n * Specifies that the corresponding type should be ignored during type inference.\n */\n@Target(AnnotationTarget.TYPE)\n@Retention(AnnotationRetention.BINARY)\ninternal annotation class NoInfer\n\n/**\n * Specifies that the constraint built for the type during type inference should be an equality one.\n */\n@Target(AnnotationTarget.TYPE)\n@Retention(AnnotationRetention.BINARY)\ninternal annotation class Exact\n\n/**\n * Specifies that a corresponding member has the lowest priority in overload resolution.\n */\n@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY, AnnotationTarget.CONSTRUCTOR)\n@Retention(AnnotationRetention.BINARY)\ninternal annotation class LowPriorityInOverloadResolution\n\n/**\n * Specifies that the corresponding member has the highest priority in overload resolution. Effectively this means that\n * an extension annotated with this annotation will win in overload resolution over a member with the same signature.\n */\n@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY)\n@Retention(AnnotationRetention.BINARY)\ninternal annotation class HidesMembers\n\n/**\n * The value of this type parameter should be mentioned in input types (argument types, receiver type or expected type).\n */\n@Target(AnnotationTarget.TYPE_PARAMETER)\n@Retention(AnnotationRetention.BINARY)\ninternal annotation class OnlyInputTypes\n\n/**\n * Specifies that this function should not be called directly without inlining\n */\n@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY, AnnotationTarget.PROPERTY_GETTER, AnnotationTarget.PROPERTY_SETTER)\n@Retention(AnnotationRetention.BINARY)\ninternal annotation class InlineOnly\n\n/**\n * Specifies that this declaration can have dynamic receiver type.\n */\n@Target(AnnotationTarget.FUNCTION,

AnnotationTarget.PROPERTY)\n@Retention(AnnotationRetention.BINARY)\ninternal annotation class DynamicExtension\n\n/**\n * The value of this parameter should be a property reference expression (`this::foo`), referencing a `lateinit` property,\n * the backing field of which is accessible at the point where the corresponding argument is passed.\n */\n@Target(AnnotationTarget.VALUE_PARAMETER)\n@Retention(AnnotationRetention.BINARY)\n@SinceKotlin(\"1.2\")\ninternal annotation class AccessibleLateinitPropertyLiteral\n\n/**\n * Specifies that this declaration is only completely supported since the specified version.\n *\n * The Kotlin compiler of an earlier version is going to report a diagnostic on usages of this declaration.\n * The diagnostic message can be specified with [message], or via [errorCode] (takes less space, but might not be immediately clear\n * to the user). The diagnostic severity can be specified with [level]: WARNING/ERROR mean that either a warning or an error\n * is going to be reported, HIDDEN means that the declaration is going to be removed from resolution completely.\n *\n * [versionKind] specifies which version should be compared with the [version] value, when compiling the usage of the annotated declaration.\n * Note that prior to 1.2, only [RequireKotlinVersionKind.LANGUAGE_VERSION] was supported, so the Kotlin compiler before 1.2 is going to\n * treat any [RequireKotlin] as if it requires the language version. Since 1.2, the Kotlin compiler supports\n * [RequireKotlinVersionKind.LANGUAGE_VERSION], [RequireKotlinVersionKind.COMPILER_VERSION] and [RequireKotlinVersionKind.API_VERSION].\n * If the actual value of [versionKind] is something different (e.g. a new version kind, added in future versions of Kotlin),\n * Kotlin 1.2 is going to ignore this [RequireKotlin] altogether, where as Kotlin before 1.2 is going to treat this as a requirement\n * on the language version.\n *\n * This annotation is erased at compile time; its arguments are stored in a more compact form in the Kotlin metadata.\n */\n@Target(AnnotationTarget.CLASS, AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY, AnnotationTarget.CONSTRUCTOR, AnnotationTarget.TYPEALIAS)\n@Retention(AnnotationRetention.SOURCE)\n@Repeatable\n@SinceKotlin(\"1.2\")\ninternal annotation class RequireKotlin(\n   val version: String,\n   val message: String = \"\",\n   val level: DeprecationLevel = DeprecationLevel.ERROR,\n   val versionKind: RequireKotlinVersionKind = RequireKotlinVersionKind.LANGUAGE_VERSION,\n   val errorCode: Int = -1\n)\n\n/**\n * The kind of the version that is required by [RequireKotlin].\n */\n@SinceKotlin(\"1.2\")\ninternal enum class RequireKotlinVersionKind {\n   LANGUAGE_VERSION,\n   COMPILER_VERSION,\n   API_VERSION,\n}\n\n/**\n * Specifies that this declaration is a part of special DSL, used for constructing function's contract.\n */\n@Retention(AnnotationRetention.BINARY)\n@SinceKotlin(\"1.2\")\ninternal annotation class ContractsDsl\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.properties\n\nimport kotlin.reflect.KProperty\n\n/**\n * Standard property delegates.\n */\npublic object Delegates {\n   /**\n   * Returns a property delegate for a read/write property with a non-`null` value that is initialized not during\n   * object construction time but at a later time. Trying to read the property before the initial value has been\n   * assigned results in an exception.\n   *\n   * @sample samples.properties.Delegates.notNullDelegate\n   */\n   public fun <T : Any> notNull(): ReadWriteProperty<Any?, T> = NotNullVar()\n\n   /**\n   * Returns a property delegate for a read/write property that calls a specified callback function when changed.\n   * @param initialValue the initial value of the property.\n   * @param onChange the callback which is called after the change of the property is made. The value of the property\n   * has already been changed when this callback is invoked.\n   *\n   * @sample samples.properties.Delegates.observableDelegate\n   */\n   public inline fun <T> observable(initialValue: T, crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Unit):\n   ReadWriteProperty<Any?, T> =\n   object : ObservableProperty<T>(initialValue) {\n   override fun afterChange(property: KProperty<*>, oldValue: T, newValue: T) = onChange(property, oldValue, newValue)\n   }\n\n   /**\n   * Returns a property delegate for a read/write property that calls a specified callback function when changed,\n   * allowing the callback to veto the modification.\n   * @param initialValue the initial value of the property.\n   * @param onChange the callback which is called before a change to the property value is attempted.\n   * The value of the property hasn't been changed yet, when this callback is invoked.\n   * If the callback returns

`true` the value of the property is being set to the new value,\n * and if the callback returns `false` the new value is discarded and the property remains its old value.\n *\n * @sample samples.properties.Delegates.vetoableDelegate\n * @sample samples.properties.Delegates.throwVetoableDelegate\n */\n public inline fun <T> vetoable(initialValue: T, crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Boolean):\n ReadWriteProperty<Any?, T> =\n object : ObservableProperty<T>(initialValue) {\n override fun beforeChange(property: KProperty<*>, oldValue: T, newValue: T): Boolean = onChange(property, oldValue, newValue)\n }\n\n}\n\n\nprivate class NotNullVar<T : Any>() : ReadWriteProperty<Any?, T> {\n private var value: T? = null\n\n public override fun getValue(thisRef: Any?, property: KProperty<*>): T {\n return value ?: throw IllegalStateException(\"Property ${property.name} should be initialized before get.\")\n }\n\n public override fun setValue(thisRef: Any?, property: KProperty<*>, value: T) {\n this.value = value\n }\n}\n\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.properties\n\nimport kotlin.reflect.KProperty\n\n/**\n * Base interface that can be used for implementing property delegates of read-only properties.\n *\n * This is provided only for convenience; you don't have to extend this interface\n * as long as your property delegate has methods with the same signatures.\n *\n * @param T the type of object which owns the delegated property.\n * @param V the type of the property value.\n */\npublic fun interface ReadOnlyProperty<in T, out V> {\n /**\n * Returns the value of the property for the given object.\n * @param thisRef the object for which the value is requested.\n * @param property the metadata for the property.\n * @return the property value.\n */\n public operator fun getValue(thisRef: T, property: KProperty<*>): V\n}\n\n/**\n * Base interface that can be used for implementing property delegates of read-write properties.\n *\n * This is provided only for convenience; you don't have to extend this interface\n * as long as your property delegate has methods with the same signatures.\n *\n * @param T the type of object which owns the delegated property.\n * @param V the type of the property value.\n */\npublic interface ReadWriteProperty<in T, V> : ReadOnlyProperty<T, V> {\n /**\n * Returns the value of the property for the given object.\n * @param thisRef the object for which the value is requested.\n * @param property the metadata for the property.\n * @return the property value.\n */\n public override operator fun getValue(thisRef: T, property: KProperty<*>): V\n\n /**\n * Sets the value of the property for the given object.\n * @param thisRef the object for which the value is requested.\n * @param property the metadata for the property.\n * @param value the value to set.\n */\n public operator fun setValue(thisRef: T, property: KProperty<*>, value: V)\n}\n\n/**\n * Base interface that can be used for implementing property delegate providers.\n *\n * This is provided only for convenience; you don't have to extend this interface\n * as long as your delegate provider has a method with the same signature.\n *\n * @param T the type of object which owns the delegated property.\n * @param D the type of property delegates this provider provides.\n */\n@SinceKotlin(\"1.4\")\npublic fun interface PropertyDelegateProvider<in T, out D> {\n /**\n * Returns the delegate of the property for the given object.\n *\n * This function can be used to extend the logic of creating the object (e.g. perform validation checks)\n * to which the property implementation is delegated.\n *\n * @param thisRef the object for which property delegate is requested.\n * @param property the metadata for the property.\n * @return the property delegate.\n */\n public operator fun provideDelegate(thisRef: T, property: KProperty<*>): D\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.properties\n\nimport kotlin.reflect.KProperty\n\n/**\n * Implements the core logic of a property delegate for a read/write property that calls callback functions when changed.\n * @param initialValue the initial value of the property.\n */\npublic abstract class ObservableProperty<V>(initialValue: V) : ReadWriteProperty<Any?, V> {\n private var value = initialValue\n\n /**\n * The callback which is called before a change to the property value is attempted.\n * The value of the property hasn't been changed yet, when this callback is invoked.\n * If the callback returns `true` the value of the property is being set to the new value,\n * and if the callback returns `false` the new value is discarded and the

property remains its old value.\n   */\n   protected open fun beforeChange(property: KProperty<*>, oldValue: V, newValue: V): Boolean = true\n\n   /**\n   * The callback which is called after the change of the property is made. The value of the property\n   * has already been changed when this callback is invoked.\n   */\n   protected open fun afterChange(property: KProperty<*>, oldValue: V, newValue: V): Unit {}\n\n   public override fun getValue(thisRef: Any?, property: KProperty<*>): V {\n   return value\n   }\n\n   public override fun setValue(thisRef: Any?, property: KProperty<*>, value: V) {\n   val oldValue = this.value\n   if (!beforeChange(property, oldValue, value)) {\n   return\n   }\n   this.value = value\n   afterChange(property, oldValue, value)\n   }\n}","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:Suppress(\"PackageDirectoryMismatch\")\npackage kotlin\n\nimport kotlin.reflect.*\n\n/**\n * An extension operator that allows delegating a read-only property of type [V]\n * to a property reference to a property of type [V] or its subtype.\n *\n * @receiver A property reference to a read-only or mutable property of type [V] or its subtype.\n * The reference is without a receiver, i.e. it either references a top-level property or\n * has the receiver bound to it.\n *\n * Example:\n *\n * ```\n * class Login(val username: String)\n * val defaultLogin = Login(\"Admin\")\n * val defaultUsername by defaultLogin::username\n * // equivalent to\n * val defaultUserName get() = defaultLogin.username\n * ```\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <V> KProperty0<V>.getValue(thisRef: Any?, property: KProperty<*>): V {\n   return get()\n}\n\n/**\n * An extension operator that allows delegating a mutable property of type [V]\n * to a property reference to a mutable property of the same type [V].\n *\n * @receiver A property reference to a mutable property of type [V].\n * The reference is without a receiver, i.e. it either references a top-level property or\n * has the receiver bound to it.\n *\n * Example:\n *\n * ```\n * class Login(val username: String, var incorrectAttemptCounter: Int = 0)\n * val defaultLogin = Login(\"Admin\")\n * var defaultLoginAttempts by defaultLogin::incorrectAttemptCounter\n * // equivalent to\n * var defaultLoginAttempts: Int\n *   get() = defaultLogin.incorrectAttemptCounter\n *   set(value) { defaultLogin.incorrectAttemptCounter = value }\n * ```\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <V> KMutableProperty0<V>.setValue(thisRef: Any?, property: KProperty<*>, value: V) {\n   set(value)\n}\n\n\n/**\n * An extension operator that allows delegating a read-only member or extension property of type [V]\n * to a property reference to a member or extension property of type [V] or its subtype.\n *\n * @receiver A property reference to a read-only or mutable property of type [V] or its subtype.\n * The reference has an unbound receiver of type [T].\n *\n * Example:\n *\n * ```\n * class Login(val username: String)\n * val Login.user by Login::username\n * // equivalent to\n * val Login.user get() = this.username\n * ```\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <T, V> KProperty1<T, V>.getValue(thisRef: T, property: KProperty<*>): V {\n   return get(thisRef)\n}\n\n/**\n * An extension operator that allows delegating a mutable member or extension property of type [V]\n * to a property reference to a member or extension mutable property of the same type [V].\n *\n * @receiver A property reference to a read-only or mutable property of type [V] or its subtype.\n * The reference has an unbound receiver of type [T].\n *\n * Example:\n *\n * ```\n * class Login(val username: String, var incorrectAttemptCounter: Int)\n * var Login.attempts by Login::incorrectAttemptCounter\n * // equivalent to\n * var Login.attempts: Int\n *   get() = this.incorrectAttemptCounter\n *   set(value) { this.incorrectAttemptCounter = value }\n * ```\n */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <T, V> KMutableProperty1<T, V>.setValue(thisRef: T, property: KProperty<*>, value: V) {\n   set(thisRef, value)\n}","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.random\n\nimport kotlin.math.nextDown\n\n/**\n * An abstract class that is implemented by random number generator algorithms.\n *\n * The companion object [Random.Default] is the default instance of [Random].\n *\n * To get a seeded instance of random generator use [Random] function.\n *\n * @sample samples.random.Randoms.defaultRandom\n */\n@SinceKotlin(\"1.3\")\npublic abstract class Random {\n\n   /**\n   * Gets the next random [bitCount] number

```
of bits.\n     *\n     * Generates an `Int` whose lower [bitCount] bits are filled with random values and the remaining upper bits are zero.\n     *\n     * @param bitCount number of bits to generate, must be in range 0..32, otherwise the behavior is unspecified.\n     *\n     * @sample samples.random.Randoms.nextBits\n     */\n    public abstract fun nextBits(bitCount: Int): Int\n\n    /**\n     * Gets the next random `Int` from the random number generator.\n     *\n     * Generates an `Int` random value uniformly distributed between `Int.MIN_VALUE` and `Int.MAX_VALUE` (inclusive).\n     *\n     * @sample samples.random.Randoms.nextInt\n     */\n    public open fun nextInt(): Int = nextBits(32)\n\n    /**\n     * Gets the next random non-negative `Int` from the random number generator less than the specified [until] bound.\n     *\n     * Generates an `Int` random value uniformly distributed between `0` (inclusive) and the specified [until] bound (exclusive).\n     *\n     * @param until must be positive.\n     *\n     * @throws IllegalArgumentException if [until] is negative or zero.\n     *\n     * @sample samples.random.Randoms.nextIntFromUntil\n     */\n    public open fun nextInt(until: Int): Int = nextInt(0, until)\n\n    /**\n     * Gets the next random `Int` from the random number generator in the specified range.\n     *\n     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.\n     *\n     * @throws IllegalArgumentException if [from] is greater than or equal to [until].\n     *\n     * @sample samples.random.Randoms.nextIntFromUntil\n     */\n    public open fun nextInt(from: Int, until: Int): Int {\n        checkRangeBounds(from, until)\n        val n = until - from\n        if (n > 0 || n == Int.MIN_VALUE) {\n            val rnd = if (n and -n == n) {\n                val bitCount = fastLog2(n)\n                nextBits(bitCount)\n            } else {\n                var v: Int\n                do {\n                    val bits = nextInt().ushr(1)\n                    v = bits % n\n                } while (bits - v + (n - 1) < 0)\n                v\n            }\n            return from + rnd\n        } else {\n            while (true) {\n                val rnd = nextInt()\n                if (rnd in from until until) return rnd\n            }\n        }\n    }\n\n    /**\n     * Gets the next random `Long` from the random number generator.\n     *\n     * Generates a `Long` random value uniformly distributed between `Long.MIN_VALUE` and `Long.MAX_VALUE` (inclusive).\n     *\n     * @sample samples.random.Randoms.nextLong\n     */\n    public open fun nextLong(): Long = nextInt().toLong().shl(32) + nextInt()\n\n    /**\n     * Gets the next random non-negative `Long` from the random number generator less than the specified [until] bound.\n     *\n     * Generates a `Long` random value uniformly distributed between `0` (inclusive) and the specified [until] bound (exclusive).\n     *\n     * @param until must be positive.\n     *\n     * @throws IllegalArgumentException if [until] is negative or zero.\n     *\n     * @sample samples.random.Randoms.nextLongFromUntil\n     */\n    public open fun nextLong(until: Long): Long = nextLong(0, until)\n\n    /**\n     * Gets the next random `Long` from the random number generator in the specified range.\n     *\n     * Generates a `Long` random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.\n     *\n     * @throws IllegalArgumentException if [from] is greater than or equal to [until].\n     *\n     * @sample samples.random.Randoms.nextLongFromUntil\n     */\n    public open fun nextLong(from: Long, until: Long): Long {\n        checkRangeBounds(from, until)\n        val n = until - from\n        if (n > 0) {\n            val rnd: Long\n            if (n and -n == n) {\n                val nLow = n.toInt()\n                val nHigh = (n ushr 32).toInt()\n                rnd = when {\n                    nLow != 0 -> {\n                        val bitCount = fastLog2(nLow)\n                        // toUInt().toLong()\n                        nextBits(bitCount).toLong() and 0xFFFF_FFFF\n                    }\n                    nHigh == 1 ->\n                        // toUInt().toLong()\n                        nextInt().toLong() and 0xFFFF_FFFF\n                    else -> {\n                        val bitCount = fastLog2(nHigh)\n                        nextBits(bitCount).toLong().shl(32) + (nextInt().toLong() and 0xFFFF_FFFF)\n                    }\n                }\n            } else {\n                var v: Long\n                do {\n                    val bits = nextLong().ushr(1)\n                    v = bits % n\n                } while (bits - v + (n - 1) < 0)\n                rnd = v\n            }\n            return from + rnd\n        } else {\n            while (true) {\n                val rnd = nextLong()\n                if (rnd in from until until) return rnd\n            }\n        }\n    }\n\n    /**\n     * Gets the next random [Boolean] value.\n     *\n     * @sample samples.random.Randoms.nextBoolean\n     */\n    public open fun nextBoolean(): Boolean = nextBits(1) != 0\n\n    /**\n     * Gets the next random [Double] value uniformly distributed between 0 (inclusive) and 1 (exclusive).\n     *\n     * @sample samples.random.Randoms.nextDouble\n     */\n    public open fun nextDouble(): Double = doubleFromParts(nextBits(26), nextBits(27))\n\n    /**\n     * Gets the next random non-negative `Double` from the random number generator less than the specified [until] bound.\n     *\n     * Generates a `Double` random value
```

```
uniformly distributed between 0 (inclusive) and [until] (exclusive).\n     *\n     * @throws IllegalArgumentException if [until] is negative or zero.\n     *\n     * @sample samples.random.Randoms.nextDoubleFromUntil\n     */\n public open fun nextDouble(until: Double): Double = nextDouble(0.0, until)\n\n    /**\n     * Gets the next random `Double` from the random number generator in the specified range.\n     *\n     * Generates a `Double` random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.\n     *\n     * [from] and [until] must be finite otherwise the behavior is unspecified.\n     *\n     * @throws IllegalArgumentException if [from] is greater than or equal to [until].\n     *\n     * @sample samples.random.Randoms.nextDoubleFromUntil\n     */\n    public open fun nextDouble(from: Double, until: Double): Double {\n        checkRangeBounds(from, until)\n        val size = until - from\n        val r = if (size.isInfinite() && from.isFinite() && until.isFinite()) {\n            val r1 = nextDouble() * (until / 2 - from / 2)\n            from + r1 + r1\n        } else {\n            from + nextDouble() * size\n        }\n        return if (r >= until) until.nextDown() else r\n    }\n\n    /**\n     * Gets the next random [Float] value uniformly distributed between 0 (inclusive) and 1 (exclusive).\n     *\n     * @sample samples.random.Randoms.nextFloat\n     */\n    public open fun nextFloat(): Float = nextBits(24) / (1 shl 24).toFloat()\n\n    /**\n     * Fills a subrange of the specified byte [array] starting from [fromIndex] inclusive and ending [toIndex] exclusive\n     * with random bytes.\n     *\n     * @return [array] with the subrange filled with random bytes.\n     *\n     * @sample samples.random.Randoms.nextBytes\n     */\n    public open fun nextBytes(array: ByteArray, fromIndex: Int = 0, toIndex: Int = array.size): ByteArray {\n        require(fromIndex in 0..array.size && toIndex in 0..array.size) { \"fromIndex ($fromIndex) or toIndex ($toIndex) are out of range: 0..${array.size}.\" }\n        require(fromIndex <= toIndex) { \"fromIndex ($fromIndex) must be not greater than toIndex ($toIndex).\" }\n\n        val steps = (toIndex - fromIndex) / 4\n\n        var position = fromIndex\n        repeat(steps) {\n            val v = nextInt()\n            array[position] = v.toByte()\n            array[position + 1] = v.ushr(8).toByte()\n            array[position + 2] = v.ushr(16).toByte()\n            array[position + 3] = v.ushr(24).toByte()\n            position += 4\n        }\n\n        val remainder = toIndex - position\n        val vr = nextBits(remainder * 8)\n        for (i in 0 until remainder) {\n            array[position + i] = vr.ushr(i * 8).toByte()\n        }\n\n        return array\n    }\n\n    /**\n     * Fills the specified byte [array] with random bytes and returns it.\n     *\n     * @return [array] filled with random bytes.\n     *\n     * @sample samples.random.Randoms.nextBytes\n     */\n    public open fun nextBytes(array: ByteArray): ByteArray = nextBytes(array, 0, array.size)\n\n    /**\n     * Creates a byte array of the specified [size], filled with random bytes.\n     *\n     * @sample samples.random.Randoms.nextBytes\n     */\n    public open fun nextBytes(size: Int): ByteArray = nextBytes(ByteArray(size))\n\n\n    /**\n     * The default random number generator.\n     *\n     * On JVM this generator is thread-safe, its methods can be invoked from multiple threads.\n     *\n     * @sample samples.random.Randoms.defaultRandom\n     */\n    companion object Default : Random(), Serializable {\n        private val defaultRandom: Random = defaultPlatformRandom()\n\n        private object Serialized : Serializable {\n            private const val serialVersionUID = 0L\n\n            private fun readResolve(): Any = Random\n        }\n\n        private fun writeReplace(): Any = Serialized\n\n        override fun nextBits(bitCount: Int): Int = defaultRandom.nextBits(bitCount)\n        override fun nextInt(): Int = defaultRandom.nextInt()\n        override fun nextInt(until: Int): Int = defaultRandom.nextInt(until)\n        override fun nextInt(from: Int, until: Int): Int = defaultRandom.nextInt(from, until)\n\n        override fun nextLong(): Long = defaultRandom.nextLong()\n        override fun nextLong(until: Long): Long = defaultRandom.nextLong(until)\n        override fun nextLong(from: Long, until: Long): Long = defaultRandom.nextLong(from, until)\n\n        override fun nextBoolean(): Boolean = defaultRandom.nextBoolean()\n\n        override fun nextDouble(): Double = defaultRandom.nextDouble()\n        override fun nextDouble(until: Double): Double = defaultRandom.nextDouble(until)\n        override fun nextDouble(from: Double, until: Double): Double = defaultRandom.nextDouble(from, until)\n\n        override fun nextFloat(): Float = defaultRandom.nextFloat()\n\n        override fun nextBytes(array: ByteArray): ByteArray = defaultRandom.nextBytes(array)\n        override fun nextBytes(size: Int): ByteArray = defaultRandom.nextBytes(size)\n        override fun nextBytes(array: ByteArray, fromIndex: Int, toIndex: Int): ByteArray =\n            defaultRandom.nextBytes(array, fromIndex, toIndex)\n    }\n}\n\n/**\n * Returns a repeatable random number generator seeded with the given [seed] `Int` value.\n *\n * Two generators with the same seed
```

produce the same sequence of values within the same version of Kotlin runtime.\n *\n * *Note:* Future versions of Kotlin may change the algorithm of this seeded number generator so that it will return\n * a sequence of values different from the current one for a given seed.\n *\n * On JVM the returned generator is NOT thread-safe. Do not invoke it from multiple threads without proper synchronization.\n *\n * @sample samples.random.Randoms.seededRandom\n */\n@SinceKotlin(\"1.3\")\npublic fun Random(seed: Int): Random = XorWowRandom(seed, seed.shr(31))\n\n/**\n * Returns a repeatable random number generator seeded with the given [seed] `Long` value.\n *\n * Two generators with the same seed produce the same sequence of values within the same version of Kotlin runtime.\n *\n * *Note:* Future versions of Kotlin may change the algorithm of this seeded number generator so that it will return\n * a sequence of values different from the current one for a given seed.\n *\n * On JVM the returned generator is NOT thread-safe. Do not invoke it from multiple threads without proper synchronization.\n *\n * @sample samples.random.Randoms.seededRandom\n */\n@SinceKotlin(\"1.3\")\npublic fun Random(seed: Long): Random = XorWowRandom(seed.toInt(), seed.shr(32).toInt())\n\n\n/**\n * Gets the next random `Int` from the random number generator in the specified [range].\n *\n * Generates an `Int` random value uniformly distributed in the specified [range]:\n * from `range.start` inclusive to `range.endInclusive` inclusive.\n *\n * @throws IllegalArgumentException if [range] is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun Random.nextInt(range: IntRange): Int = when {\n   range.isEmpty() -> throw IllegalArgumentException(\"Cannot get random in empty range: $range\")\n   range.last < Int.MAX_VALUE -> nextInt(range.first, range.last + 1)\n   range.first > Int.MIN_VALUE -> nextInt(range.first - 1, range.last) + 1\n   else -> nextInt()\n}\n\n/**\n * Gets the next random `Long` from the random number generator in the specified [range].\n *\n * Generates a `Long` random value uniformly distributed in the specified [range]:\n * from `range.start` inclusive to `range.endInclusive` inclusive.\n *\n * @throws IllegalArgumentException if [range] is empty.\n */\n@SinceKotlin(\"1.3\")\npublic fun Random.nextLong(range: LongRange): Long = when {\n   range.isEmpty() -> throw IllegalArgumentException(\"Cannot get random in empty range: $range\")\n   range.last < Long.MAX_VALUE -> nextLong(range.first, range.last + 1)\n   range.first > Long.MIN_VALUE -> nextLong(range.first - 1, range.last) + 1\n   else -> nextLong()\n}\n\n\ninternal expect fun defaultPlatformRandom(): Random\ninternal expect fun doubleFromParts(hi26: Int, low27: Int): Double\n\ninternal fun fastLog2(value: Int): Int = 31 - value.countLeadingZeroBits()\n\n/** Takes upper [bitCount] bits (0..32) from this number. */\ninternal fun Int.takeUpperBits(bitCount: Int): Int =\n   this.ushr(32 - bitCount) and (-bitCount).shr(31)\n\ninternal fun checkRangeBounds(from: Int, until: Int) = require(until > from) { boundsErrorMessage(from, until) }\ninternal fun checkRangeBounds(from: Long, until: Long) = require(until > from) { boundsErrorMessage(from, until) }\ninternal fun checkRangeBounds(from: Double, until: Double) = require(until > from) { boundsErrorMessage(from, until) }\n\ninternal fun boundsErrorMessage(from: Any, until: Any) = \"Random range is empty: [$from, $until).\"\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.random\n\n\n/**\n * Gets the next random [UInt] from the random number generator.\n *\n * Generates a [UInt] random value uniformly distributed between [UInt.MIN_VALUE] and [UInt.MAX_VALUE] (inclusive).\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Random.nextUInt(): UInt = nextInt().toUInt()\n\n/**\n * Gets the next random [UInt] from the random number generator less than the specified [until] bound.\n *\n * Generates a [UInt] random value uniformly distributed between `0` (inclusive) and the specified [until] bound (exclusive).\n *\n * @throws IllegalArgumentException if [until] is zero.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Random.nextUInt(until: UInt): UInt = nextUInt(0u, until)\n\n/**\n * Gets the next random [UInt] from the random number generator in the specified range.\n *\n * Generates a [UInt] random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.\n *\n * @throws IllegalArgumentException if [from] is greater than or equal to [until].\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Random.nextUInt(from: UInt, until: UInt): UInt {\n   checkUIntRangeBounds(from, until)\n\n   val signedFrom =

from.toInt() xor Int.MIN_VALUE\n   val signedUntil = until.toInt() xor Int.MIN_VALUE\n\n   val signedResult = nextInt(signedFrom, signedUntil) xor Int.MIN_VALUE\n   return signedResult.toUInt()\n}\n\n/**\n * Gets the next random [UInt] from the random number generator in the specified [range].\n *\n * Generates a [UInt] random value uniformly distributed in the specified [range]:\n * from `range.start` inclusive to `range.endInclusive` inclusive.\n *\n * @throws IllegalArgumentException if [range] is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Random.nextUInt(range: UIntRange): UInt = when {\n   range.isEmpty() -> throw IllegalArgumentException(\"Cannot get random in empty range: $range\")\n   range.last < UInt.MAX_VALUE -> nextUInt(range.first, range.last + 1u)\n   range.first > UInt.MIN_VALUE -> nextUInt(range.first - 1u, range.last) + 1u\n   else -> nextUInt()\n}\n\n/**\n * Gets the next random [ULong] from the random number generator.\n *\n * Generates a [ULong] random value uniformly distributed between [ULong.MIN_VALUE] and [ULong.MAX_VALUE] (inclusive).\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Random.nextULong(): ULong = nextLong().toULong()\n\n/**\n * Gets the next random [ULong] from the random number generator less than the specified [until] bound.\n *\n * Generates a [ULong] random value uniformly distributed between `0` (inclusive) and the specified [until] bound (exclusive).\n *\n * @throws IllegalArgumentException if [until] is zero.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Random.nextULong(until: ULong): ULong = nextULong(0uL, until)\n\n/**\n * Gets the next random [ULong] from the random number generator in the specified range.\n *\n * Generates a [ULong] random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.\n *\n * @throws IllegalArgumentException if [from] is greater than or equal to [until].\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Random.nextULong(from: ULong, until: ULong): ULong {\n   checkULongRangeBounds(from, until)\n\n   val signedFrom = from.toLong() xor Long.MIN_VALUE\n   val signedUntil = until.toLong() xor Long.MIN_VALUE\n\n   val signedResult = nextLong(signedFrom, signedUntil) xor Long.MIN_VALUE\n   return signedResult.toULong()\n}\n\n/**\n * Gets the next random [ULong] from the random number generator in the specified [range].\n *\n * Generates a [ULong] random value uniformly distributed in the specified [range]:\n * from `range.start` inclusive to `range.endInclusive` inclusive.\n *\n * @throws IllegalArgumentException if [range] is empty.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun Random.nextULong(range: ULongRange): ULong = when {\n   range.isEmpty() -> throw IllegalArgumentException(\"Cannot get random in empty range: $range\")\n   range.last < ULong.MAX_VALUE -> nextULong(range.first, range.last + 1u)\n   range.first > ULong.MIN_VALUE -> nextULong(range.first - 1u, range.last) + 1u\n   else -> nextULong()\n}\n\n/**\n * Fills the specified unsigned byte [array] with random bytes and returns it.\n *\n * @return [array] filled with random bytes.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Random.nextUBytes(array: UByteArray): UByteArray {\n   nextBytes(array.asByteArray())\n   return array\n}\n\n/**\n * Creates an unsigned byte array of the specified [size], filled with random bytes.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Random.nextUBytes(size: Int): UByteArray = nextBytes(size).asUByteArray()\n\n/**\n * Fills a subrange of the specified `UByte` [array] starting from [fromIndex] inclusive and ending [toIndex] exclusive with random UBytes.\n *\n * @return [array] with the subrange filled with random bytes.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalUnsignedTypes\npublic fun Random.nextUBytes(array: UByteArray, fromIndex: Int = 0, toIndex: Int = array.size): UByteArray {\n   nextBytes(array.asByteArray(), fromIndex, toIndex)\n   return array\n}\n\n\ninternal fun checkUIntRangeBounds(from: UInt, until: UInt) = require(until > from) { boundsErrorMessage(from, until) }\ninternal fun checkULongRangeBounds(from: ULong, until: ULong) = require(until > from) { boundsErrorMessage(from, until) }\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.random\n\n/**\n * Random number generator,

using Marsaglia's \"xorwow\" algorithm\n *\n * Cycles after 2^192 - 2^32 repetitions.\n *\n * For more details, see Marsaglia, George (July 2003). \"Xorshift RNGs\". Journal of Statistical Software. 8 (14). doi:10.18637/jss.v008.i14\n *\n * Available at https://www.jstatsoft.org/v08/i14/paper\n *\n */\ninternal class XorWowRandom internal constructor(\n   private var x: Int,\n   private var y: Int,\n   private var z: Int,\n   private var w: Int,\n   private var v: Int,\n   private var addend: Int\n) : Random(), Serializable {\n\n   internal constructor(seed1: Int, seed2: Int) :\n   this(seed1, seed2, 0, 0, seed1.inv(), (seed1 shl 10) xor (seed2 ushr 4))\n\n   init {\n   require((x or y or z or w or v) != 0) { \"Initial state must have at least one non-zero element.\" }\n\n   // some trivial seeds can produce several values with zeroes in upper bits, so we discard first 64\n   repeat(64) { nextInt() }\n   }\n\n   override fun nextInt(): Int {\n   // Equivalent to the xorxow algorithm\n   // From Marsaglia, G. 2003. Xorshift RNGs. J. Statis. Soft. 8, 14, p. 5\n   var t = x\n   t = t xor (t ushr 2)\n   x = y\n   y = z\n   z = w\n   val v0 = v\n   w = v0\n   t = (t xor (t shl 1)) xor v0 xor (v0 shl 4)\n   v = t\n   addend += 362437\n   return t + addend\n   }\n\n   override fun nextBits(bitCount: Int): Int =\n   nextInt().takeUpperBits(bitCount)\n\n   private companion object {\n   private const val serialVersionUID: Long = 0L\n   }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"RangesKt\")\n\npackage kotlin.ranges\n\n/**\n * Represents a range of [Comparable] values.\n */\nprivate open class ComparableRange<T : Comparable<T>>(\n   override val start: T,\n   override val endInclusive: T\n) : ClosedRange<T> {\n\n   override fun equals(other: Any?): Boolean {\n   return other is ComparableRange<*> && (isEmpty() && other.isEmpty() ||\n   start == other.start && endInclusive == other.endInclusive)\n   }\n\n   override fun hashCode(): Int {\n   return if (isEmpty()) -1 else 31 * start.hashCode() + endInclusive.hashCode()\n   }\n\n   override fun toString(): String = \"$start..$endInclusive\"\n}\n\n/**\n * Creates a range from this [Comparable] value to the specified [that] value.\n *\n * This value needs to be smaller than or equal to [that] value, otherwise the returned range will be empty.\n * @sample samples.ranges.Ranges.rangeFromComparable\n */\npublic operator fun <T : Comparable<T>> T.rangeTo(that: T): ClosedRange<T> = ComparableRange(this, that)\n\n\n/**\n * Represents a range of floating point numbers.\n * Extends [ClosedRange] interface providing custom operation [lessThanOrEquals] for comparing values of range domain type.\n *\n * This interface is implemented by floating point ranges returned by [Float.rangeTo] and [Double.rangeTo] operators to\n * achieve IEEE-754 comparison order instead of total order of floating point numbers.\n */\n@SinceKotlin(\"1.1\")\npublic interface ClosedFloatingPointRange<T : Comparable<T>> : ClosedRange<T> {\n   override fun contains(value: T): Boolean = lessThanOrEquals(start, value) && lessThanOrEquals(value, endInclusive)\n   override fun isEmpty(): Boolean = !lessThanOrEquals(start, endInclusive)\n\n   /**\n   * Compares two values of range domain type and returns true if first is less than or equal to second.\n   */\n   fun lessThanOrEquals(a: T, b: T): Boolean\n}\n\n\n/**\n * A closed range of values of type `Double`.\n *\n * Numbers are compared with the ends of this range according to IEEE-754.\n */\nprivate class ClosedDoubleRange(\n   start: Double,\n   endInclusive: Double\n) : ClosedFloatingPointRange<Double> {\n   private val _start = start\n   private val _endInclusive = endInclusive\n   override val start: Double get() = _start\n   override val endInclusive: Double get() = _endInclusive\n\n   override fun lessThanOrEquals(a: Double, b: Double): Boolean = a <= b\n\n   override fun contains(value: Double): Boolean = value >= _start && value <= _endInclusive\n   override fun isEmpty(): Boolean = !(_start <= _endInclusive)\n\n   override fun equals(other: Any?): Boolean {\n   return other is ClosedDoubleRange && (isEmpty() && other.isEmpty() ||\n   _start == other._start && _endInclusive == other._endInclusive)\n   }\n\n   override fun hashCode(): Int {\n   return if (isEmpty()) -1 else 31 * _start.hashCode() + _endInclusive.hashCode()\n   }\n\n   override fun toString(): String = \"$_start..$_endInclusive\"\n}\n\n/**\n * Creates a range from this [Double] value to the specified [that] value.\n *\n * Numbers are compared with the ends of this range according to IEEE-754.\n * @sample samples.ranges.Ranges.rangeFromDouble\n */\n@SinceKotlin(\"1.1\")\npublic operator fun Double.rangeTo(that: Double): ClosedFloatingPointRange<Double> = ClosedDoubleRange(this, that)\n\n\n/**\n * A closed range of values of type `Float`.\n *\n * Numbers are compared with the ends of this range according to IEEE-754.\n */\nprivate class ClosedFloatRange(\n   start: Float,\n   endInclusive: Float\n) :

ClosedFloatingPointRange<Float> {\n   private val _start = start\n   private val _endInclusive = endInclusive\n   override val start: Float get() = _start\n   override val endInclusive: Float get() = _endInclusive\n\n   override fun lessThanOrEquals(a: Float, b: Float): Boolean = a <= b\n\n   override fun contains(value: Float): Boolean = value >= _start && value <= _endInclusive\n   override fun isEmpty(): Boolean = !(_start <= _endInclusive)\n\n   override fun equals(other: Any?): Boolean {\n   return other is ClosedFloatRange && (isEmpty() && other.isEmpty() ||\n   _start == other._start && _endInclusive == other._endInclusive)\n   }\n\n   override fun hashCode(): Int {\n   return if (isEmpty()) -1 else 31 * _start.hashCode() + _endInclusive.hashCode()\n   }\n\n   override fun toString(): String = \"$_start..$_endInclusive\"\n}\n\n/**\n * Creates a range from this [Float] value to the specified [that] value.\n *\n * Numbers are compared with the ends of this range according to IEEE-754.\n * @sample samples.ranges.Ranges.rangeFromFloat\n */\n@SinceKotlin(\"1.1\")\npublic operator fun Float.rangeTo(that: Float): ClosedFloatingPointRange<Float> = ClosedFloatRange(this, that)\n\n\n/**\n * Returns `true` if this iterable range contains the specified [element].\n *\n * Always returns `false` if the [element] is `null`.\n */\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline operator fun <T, R> R.contains(element: T?): Boolean where T : Any, R : Iterable<T>, R : ClosedRange<T> =\n   element != null && contains(element)\n\n\ninternal fun checkStepIsPositive(isPositive: Boolean, step: Number) {\n   if (!isPositive) throw IllegalArgumentException(\"Step must be positive, was: $step.\")\n}\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmName(\"KClasses\")\n@file:Suppress(\"UNCHECKED_CAST\")\n\npackage kotlin.reflect\n\nimport kotlin.internal.LowPriorityInOverloadResolution\n\n/**\n * Casts the given [value] to the class represented by this [KClass] object.\n * Throws an exception if the value is `null` or if it is not an instance of this class.\n *\n * This is an experimental function that behaves as a similar function from kotlin.reflect.full on JVM.\n *\n * @see [KClass.isInstance]\n * @see [KClass.safeCast]\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@LowPriorityInOverloadResolution\nfun <T : Any> KClass<T>.cast(value: Any?): T {\n   if (!isInstance(value)) throw ClassCastException(\"Value cannot be cast to $qualifiedOrSimpleName\")\n   return value as T\n}\n\n// TODO: replace with qualifiedName when it is fully supported in K/JS\ninternal expect val KClass<*>.qualifiedOrSimpleName: String?\n\n/**\n * Casts the given [value] to the class represented by this [KClass] object.\n * Returns `null` if the value is `null` or if it is not an instance of this class.\n *\n * This is an experimental function that behaves as a similar function from kotlin.reflect.full on JVM.\n *\n * @see [KClass.isInstance]\n * @see [KClass.cast]\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@LowPriorityInOverloadResolution\nfun <T : Any> KClass<T>.safeCast(value: Any?): T? {\n   return if (isInstance(value)) value as T else null\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.reflect\n\nimport kotlin.jvm.JvmField\nimport kotlin.jvm.JvmStatic\n\n\n\n/**\n * Represents a type projection. Type projection is usually the argument to another type in a type usage.\n * For example, in the type `Array<out Number>`, `out Number` is the covariant projection of the type represented by the class `Number`.\n *\n * Type projection is either the star projection, or an entity consisting of a specific type plus optional variance.\n *\n * See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/generics.html#type-projections)\n * for more information.\n */\n@SinceKotlin(\"1.1\")\npublic data class KTypeProjection constructor(\n   /**\n   * The use-site variance specified in the projection, or `null` if this is a star projection.\n   */\n   public val variance: KVariance?,\n   /**\n   * The type specified in the projection, or `null` if this is a star projection.\n   */\n   public val type: KType?\n) {\n\n   init {\n   require((variance == null) == (type == null)) {\n   if (variance == null)\n   \"Star projection must have no type specified.\"\n   else\n   \"The projection variance $variance requires type to be specified.\"\n   }\n   }\n\n   override fun toString(): String = when (variance) {\n   null -> \"*\"\n   KVariance.INVARIANT -> type.toString()\n   KVariance.IN -> \"in $type\"\n   KVariance.OUT -> \"out $type\"\n   }\n\n   public companion object {\n   // provided for compiler access\n   @JvmField\n   @PublishedApi\n   internal val star: KTypeProjection =

KTypeProjection(null, null)\n\n /**\n * Star projection, denoted by the `*` character.\n * For example, in the type `KClass<*>`, `*` is the star projection.\n * See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/generics.html#star-projections)\n * for more information.\n */\n public val STAR: KTypeProjection get() = star\n\n /**\n * Creates an invariant projection of a given type. Invariant projection is just the type itself,\n * without any use-site variance modifiers applied to it.\n * For example, in the type `Set<String>`, `String` is an invariant projection of the type represented by the class `String`.\n */\n @JvmStatic\n public fun invariant(type: KType): KTypeProjection =\n KTypeProjection(KVariance.INVARIANT, type)\n\n /**\n * Creates a contravariant projection of a given type, denoted by the `in` modifier applied to a type.\n * For example, in the type `MutableList<in Number>`, `in Number` is a contravariant projection of the type of class `Number`.\n */\n @JvmStatic\n public fun contravariant(type: KType): KTypeProjection =\n KTypeProjection(KVariance.IN, type)\n\n /**\n * Creates a covariant projection of a given type, denoted by the `out` modifier applied to a type.\n * For example, in the type `Array<out Number>`, `out Number` is a covariant projection of the type of class `Number`.\n */\n @JvmStatic\n public fun covariant(type: KType): KTypeProjection =\n KTypeProjection(KVariance.OUT, type)\n }\n}","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.reflect\n\n/**\n * Represents variance applied to a type parameter on the declaration site (*declaration-site variance*),\n * or to a type in a projection (*use-site variance*).\n *\n * See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/generics.html#variance)\n * for more information.\n *\n * @see [KTypeParameter.variance]\n * @see [KTypeProjection]\n */\n@SinceKotlin(\"1.1\")\nenum class KVariance {\n /**\n * The affected type parameter or type is *invariant*, which means it has no variance applied to it.\n */\n INVARIANT,\n\n /**\n * The affected type parameter or type is *contravariant*. Denoted by the `in` modifier in the source code.\n */\n IN,\n\n /**\n * The affected type parameter or type is *covariant*. Denoted by the `out` modifier in the source code.\n */\n OUT,\n}","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.reflect\n\n/**\n * Returns a runtime representation of the given reified type [T] as an instance of [KType].\n *\n * Note that on JVM, the created type has no annotations ([KType.annotations] returns an empty list)\n * even if the type in the source code is annotated. Support for type annotations might be added in a future version.\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <reified T> typeOf(): KType =\n throw UnsupportedOperationException(\"This function is implemented as an intrinsic on all supported platforms.\")\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"StringsKt\")\n\npackage kotlin.text\n\n/**\n * An object to which char sequences and values can be appended.\n */\nexpect interface Appendable {\n /**\n * Appends the specified character [value] to this Appendable and returns this instance.\n *\n * @param value the character to append.\n */\n fun append(value: Char): Appendable\n\n /**\n * Appends the specified character sequence [value] to this Appendable and returns this instance.\n *\n * @param value the character sequence to append. If [value] is `null`, then the four characters `\"null\"` are appended to this Appendable.\n */\n fun append(value: CharSequence?): Appendable\n\n /**\n * Appends a subsequence of the specified character sequence [value] to this Appendable and returns this instance.\n *\n * @param value the character sequence from which a subsequence is appended. If [value] is `null`,\n * then characters are appended as if [value] contained the four characters `\"null\"`.\n * @param startIndex the beginning (inclusive) of the subsequence to append.\n * @param endIndex the end (exclusive) of the subsequence to append.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.\n */\n fun append(value:

CharSequence?, startIndex: Int, endIndex: Int): Appendable\n}\n\n/**\n * Appends a subsequence of the specified character sequence [value] to this Appendable and returns this instance.\n *\n * @param value the character sequence from which a subsequence is appended.\n * @param startIndex the beginning (inclusive) of the subsequence to append.\n * @param endIndex the end (exclusive) of the subsequence to append.\n *\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun <T : Appendable> T.appendRange(value: CharSequence, startIndex: Int, endIndex: Int): T {\n @Suppress(\"UNCHECKED_CAST\")\n return append(value, startIndex, endIndex) as T\n}\n\n/**\n * Appends all arguments to the given [Appendable].\n */\npublic fun <T : Appendable> T.append(vararg value: CharSequence?): T {\n for (item in value)\n append(item)\n return this\n}\n\n/** Appends a line feed character (`\\n`) to this Appendable. */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun Appendable.appendLine(): Appendable = append('\\n')\n\n/** Appends value to the given Appendable and a line feed character (`\\n`) after it. */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun Appendable.appendLine(value: CharSequence?): Appendable = append(value).appendLine()\n\n/** Appends value to the given Appendable and a line feed character (`\\n`) after it. */\n@SinceKotlin(\"1.4\")\n@kotlin.internal.InlineOnly\npublic inline fun Appendable.appendLine(value: Char): Appendable = append(value).appendLine()\n\n\ninternal fun <T> Appendable.appendElement(element: T, transform: ((T) -> CharSequence)?) {\n when {\n transform != null -> append(transform(element))\n element is CharSequence? -> append(element)\n element is Char -> append(element)\n else -> append(element.toString())\n }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"StringsKt\")\n\npackage kotlin.text\n\n/**\n * Trims leading whitespace characters followed by [marginPrefix] from every line of a source string and removes\n * the first and the last lines if they are blank (notice difference blank vs empty).\n *\n * Doesn't affect a line if it doesn't contain [marginPrefix] except the first and the last blank lines.\n *\n * Doesn't preserve the original line endings.\n *\n * @param marginPrefix non-blank string, which is used as a margin delimiter. Default is `|` (pipe character).\n *\n * @sample samples.text.Strings.trimMargin\n * @see trimIndent\n * @see kotlin.text.isWhitespace\n */\npublic fun String.trimMargin(marginPrefix: String = \"|\"): String =\n replaceIndentByMargin(\"\", marginPrefix)\n\n/**\n * Detects indent by [marginPrefix] as it does [trimMargin] and replace it with [newIndent].\n *\n * @param marginPrefix non-blank string, which is used as a margin delimiter. Default is `|` (pipe character).\n */\npublic fun String.replaceIndentByMargin(newIndent: String = \"\", marginPrefix: String = \"|\"): String {\n require(marginPrefix.isNotBlank()) { \"marginPrefix must be non-blank string.\" }\n val lines = lines()\n\n return lines.reindent(length + newIndent.length * lines.size, getIndentFunction(newIndent), { line ->\n val firstNonWhitespaceIndex = line.indexOfFirst { !it.isWhitespace() }\n\n when {\n firstNonWhitespaceIndex == -1 -> null\n line.startsWith(marginPrefix, firstNonWhitespaceIndex) -> line.substring(firstNonWhitespaceIndex + marginPrefix.length)\n else -> null\n }\n })\n}\n\n/**\n * Detects a common minimal indent of all the input lines, removes it from every line and also removes the first and the last\n * lines if they are blank (notice difference blank vs empty).\n *\n * Note that blank lines do not affect the detected indent level.\n *\n * In case if there are non-blank lines with no leading whitespace characters (no indent at all) then the\n * common indent is 0, and therefore this function doesn't change the indentation.\n *\n * Doesn't preserve the original line endings.\n *\n * @sample samples.text.Strings.trimIndent\n * @see trimMargin\n * @see kotlin.text.isBlank\n */\npublic fun String.trimIndent(): String = replaceIndent(\"\")\n\n/**\n * Detects a common minimal indent like it does [trimIndent] and replaces it with the specified [newIndent].\n */\npublic fun String.replaceIndent(newIndent: String = \"\"): String {\n val lines = lines()\n\n val minCommonIndent = lines\n .filter(String::isNotBlank)\n .map(String::indentWidth)\n .minOrNull() ?: 0\n\n return lines.reindent(length + newIndent.length *

```
lines.size, getIndentFunction(newIndent), { line -> line.drop(minCommonIndent) })\n}\n\n/**\n * Prepends [indent] to every line of the original string.\n *\n * Doesn't preserve the original line endings.\n */\npublic fun String.prependIndent(indent: String = \"    \"): String =\n    lineSequence()\n        .map {\n            when {\n                it.isBlank() -> {\n                    when {\n                        it.length < indent.length -> indent\n                        else -> it\n                    }\n                }\n                else -> indent + it\n            }\n        }\n        .joinToString(\"\\n\")\n\nprivate fun String.indentWidth(): Int = indexOfFirst { !it.isWhitespace() }.let { if (it == -1) length else it }\n\nprivate fun getIndentFunction(indent: String) = when {\n    indent.isEmpty() -> { line: String -> line }\n    else -> { line: String -> indent + line }\n}\n\nprivate inline fun List<String>.reindent(\n    resultSizeEstimate: Int,\n    indentAddFunction: (String) -> String,\n    indentCutFunction: (String) -> String?\n): String {\n    val lastIndex = lastIndex\n    return mapIndexedNotNull { index, value ->\n        if ((index == 0 || index == lastIndex) && value.isBlank())\n            null\n        else\n            indentCutFunction(value)?.let(indentAddFunction) ?: value\n    }\n    .joinTo(StringBuilder(resultSizeEstimate), \"\\n\")\n        .toString()\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n/**\n * Defines names for Unicode symbols used in proper Typography.\n */\npublic object Typography {\n    /** The character &#x22; \u2013 quotation mark */\n    public const val quote: Char = '\\u0022'\n    /** The character &#x24; \u2013 dollar sign */\n    public const val dollar: Char = '\\u0024'\n    /** The character &#x26; \u2013 ampersand */\n    public const val amp: Char = '\\u0026'\n    /** The character &#x3C; \u2013 less-than sign */\n    public const val less: Char = '\\u003C'\n    /** The character &#x3E; \u2013 greater-than sign */\n    public const val greater: Char = '\\u003E'\n    /** The non-breaking space character */\n    public const val nbsp: Char = '\\u00A0'\n    /** The character &#xD7; */\n    public const val times: Char = '\\u00D7'\n    /** The character &#xA2; */\n    public const val cent: Char = '\\u00A2'\n    /** The character &#xA3; */\n    public const val pound: Char = '\\u00A3'\n    /** The character &#xA7; */\n    public const val section: Char = '\\u00A7'\n    /** The character &#xA9; */\n    public const val copyright: Char = '\\u00A9'\n    /** The character &#xAB; */\n    @SinceKotlin(\"1.6\")\n    public const val leftGuillemet: Char = '\\u00AB'\n    /** The character &#xBB; */\n    @SinceKotlin(\"1.6\")\n    public const val rightGuillemet: Char = '\\u00BB'\n    /** The character &#xAE; */\n    public const val registered: Char = '\\u00AE'\n    /** The character &#xB0; */\n    public const val degree: Char = '\\u00B0'\n    /** The character &#xB1; */\n    public const val plusMinus: Char = '\\u00B1'\n    /** The character &#xB6; */\n    public const val paragraph: Char = '\\u00B6'\n    /** The character &#xB7; */\n    public const val middleDot: Char = '\\u00B7'\n    /** The character &#xBD; */\n    public const val half: Char = '\\u00BD'\n    /** The character &#x2013; */\n    public const val ndash: Char = '\\u2013'\n    /** The character &#x2014; */\n    public const val mdash: Char = '\\u2014'\n    /** The character &#x2018; */\n    public const val leftSingleQuote: Char = '\\u2018'\n    /** The character &#x2019; */\n    public const val rightSingleQuote: Char = '\\u2019'\n    /** The character &#x201A; */\n    public const val lowSingleQuote: Char = '\\u201A'\n    /** The character &#x201C; */\n    public const val leftDoubleQuote: Char = '\\u201C'\n    /** The character &#x201D; */\n    public const val rightDoubleQuote: Char = '\\u201D'\n    /** The character &#x201E; */\n    public const val lowDoubleQuote: Char = '\\u201E'\n    /** The character &#x2020; */\n    public const val dagger: Char = '\\u2020'\n    /** The character &#x2021; */\n    public const val doubleDagger: Char = '\\u2021'\n    /** The character &#x2022; */\n    public const val bullet: Char = '\\u2022'\n    /** The character &#x2026; */\n    public const val ellipsis: Char = '\\u2026'\n    /** The character &#x2032; */\n    public const val prime: Char = '\\u2032'\n    /** The character &#x2033; */\n    public const val doublePrime: Char = '\\u2033'\n    /** The character &#x20AC; */\n    public const val euro: Char = '\\u20AC'\n    /** The character &#x2122; */\n    public const val tm: Char = '\\u2122'\n    /** The character &#x2248; */\n    public const val almostEqual: Char = '\\u2248'\n    /** The character &#x2260; */\n    public const val notEqual: Char = '\\u2260'\n    /** The character &#x2264; */\n    public const val lessOrEqual: Char = '\\u2264'\n    /** The character &#x2265; */\n    public const val greaterOrEqual: Char = '\\u2265'\n\n    /** The character &#xAB; */\n    @Deprecated(\"This constant has a typo in the name. Use leftGuillemet instead.\", ReplaceWith(\"Typography.leftGuillemet\"))\n    @DeprecatedSinceKotlin(\"1.6\")\n    public const val leftGuillemete: Char = '\\u00AB'\n\n    /** The character &#xBB; */\n    @Deprecated(\"This constant has a typo in
```

the name. Use rightGuillemet instead.\", ReplaceWith(\"Typography.rightGuillemet\"))\n @DeprecatedSinceKotlin(\"1.6\")\n public const val rightGuillemete: Char = '\\u00BB'\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n/**\n * Represents a collection of captured groups in a single match of a regular expression.\n *\n * This collection has size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.\n * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.\n *\n * An element of the collection at the particular index can be `null`,\n * if the corresponding group in the regular expression is optional and\n * there was no match captured by that group.\n */\npublic interface MatchGroupCollection : Collection<MatchGroup?> {\n\n /** Returns a group with the specified [index].\n *\n * @return An instance of [MatchGroup] if the group with the specified [index] was matched or `null` otherwise.\n *\n * Groups are indexed from 1 to the count of groups in the regular expression. A group with the index 0\n * corresponds to the entire match.\n */\n public operator fun get(index: Int): MatchGroup?\n}\n\n/**\n * Extends [MatchGroupCollection] by introducing a way to get matched groups by name, when regex supports it.\n */\n@SinceKotlin(\"1.1\")\npublic interface MatchNamedGroupCollection : MatchGroupCollection {\n /**\n * Returns a named group with the specified [name].\n * @return An instance of [MatchGroup] if the group with the specified [name] was matched or `null` otherwise.\n * @throws IllegalArgumentException if there is no group with the specified [name] defined in the regex pattern.\n * @throws UnsupportedOperationException if getting named groups isn't supported on the current platform.\n */\n public operator fun get(name: String): MatchGroup?\n}\n\n/**\n * Represents the results from a single regular expression match.\n */\npublic interface MatchResult {\n /** The range of indices in the original string where match was captured. */\n public val range: IntRange\n /** The substring from the input string captured by this match. */\n public val value: String\n /**\n * A collection of groups matched by the regular expression.\n *\n * This collection has size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.\n * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.\n */\n public val groups: MatchGroupCollection\n /**\n * A list of matched indexed group values.\n *\n * This list has size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.\n * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.\n *\n * If the group in the regular expression is optional and there were no match captured by that group,\n * corresponding item in [groupValues] is an empty string.\n *\n * @sample samples.text.Regexps.matchDestructuringToGroupValues\n */\n public val groupValues: List<String>\n\n /**\n * An instance of [MatchResult.Destructured] wrapper providing components for destructuring assignment of group values.\n *\n * component1 corresponds to the value of the first group, component2 \u2014 of the second, and so on.\n *\n * @sample samples.text.Regexps.matchDestructuringToGroupValues\n */\n public val destructured: Destructured get() = Destructured(this)\n\n /** Returns a new [MatchResult] with the results for the next match, starting at the position\n * at which the last match ended (at the character after the last matched character).\n */\n public fun next(): MatchResult?\n\n /**\n * Provides components for destructuring assignment of group values.\n *\n * [component1] corresponds to the value of the first group, [component2] \u2014 of the second, and so on.\n *\n * If the group in the regular expression is optional and there were no match captured by that group,\n * corresponding component value is an empty string.\n *\n * @sample samples.text.Regexps.matchDestructuringToGroupValues\n */\n public class Destructured internal constructor(public val match: MatchResult) {\n @kotlin.internal.InlineOnly\n public operator inline fun component1(): String = match.groupValues[1]\n @kotlin.internal.InlineOnly\n public operator inline fun component2(): String = match.groupValues[2]\n @kotlin.internal.InlineOnly\n public operator inline fun component3(): String = match.groupValues[3]\n @kotlin.internal.InlineOnly\n public operator inline fun component4(): String = match.groupValues[4]\n @kotlin.internal.InlineOnly\n public operator inline fun component5(): String = match.groupValues[5]\n @kotlin.internal.InlineOnly\n public operator inline fun component6(): String = match.groupValues[6]\n @kotlin.internal.InlineOnly\n public operator inline fun

component7(): String = match.groupValues[7]\n @kotlin.internal.InlineOnly\n public operator inline fun component8(): String = match.groupValues[8]\n @kotlin.internal.InlineOnly\n public operator inline fun component9(): String = match.groupValues[9]\n @kotlin.internal.InlineOnly\n public operator inline fun component10(): String = match.groupValues[10]\n\n /**\n * Returns destructured group values as a list of strings.\n * First value in the returned list corresponds to the value of the first group, and so on.\n *\n * @sample samples.text.Regexps.matchDestructuringToGroupValues\n */\n public fun toList(): List<String> = match.groupValues.subList(1, match.groupValues.size)\n }\n}","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass()\n@file:kotlin.jvm.JvmName(\"DurationUnitKt\")\n\npackage kotlin.time\n\n\n/**\n * The list of possible time measurement units, in which a duration can be expressed.\n *\n * The smallest time unit is [NANOSECONDS] and the largest is [DAYS], which corresponds to exactly 24 [HOURS].\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalTime::class)\npublic expect enum class DurationUnit {\n /**\n * Time unit representing one nanosecond, which is 1/1000 of a microsecond.\n */\n NANOSECONDS,\n /**\n * Time unit representing one microsecond, which is 1/1000 of a millisecond.\n */\n MICROSECONDS,\n /**\n * Time unit representing one millisecond, which is 1/1000 of a second.\n */\n MILLISECONDS,\n /**\n * Time unit representing one second.\n */\n SECONDS,\n /**\n * Time unit representing one minute.\n */\n MINUTES,\n /**\n * Time unit representing one hour.\n */\n HOURS,\n /**\n * Time unit representing one day, which is always equal to 24 hours.\n */\n DAYS;\n}\n\n/** Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified [targetUnit]. */\n@SinceKotlin(\"1.3\")\ninternal expect fun convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double\n\n// overflown result is unspecified\n@SinceKotlin(\"1.5\")\ninternal expect fun convertDurationUnitOverflow(value: Long, sourceUnit: DurationUnit, targetUnit: DurationUnit): Long\n\n// overflown result is coerced in the Long range boundaries\n@SinceKotlin(\"1.5\")\ninternal expect fun convertDurationUnit(value: Long, sourceUnit: DurationUnit, targetUnit: DurationUnit): Long\n\n\n@SinceKotlin(\"1.3\")\n@Suppress(\"REDUNDANT_ELSE_IN_WHEN\")\ninternal fun DurationUnit.shortName(): String = when (this) {\n DurationUnit.NANOSECONDS -> \"ns\"\n DurationUnit.MICROSECONDS -> \"us\"\n DurationUnit.MILLISECONDS -> \"ms\"\n DurationUnit.SECONDS -> \"s\"\n DurationUnit.MINUTES -> \"m\"\n DurationUnit.HOURS -> \"h\"\n DurationUnit.DAYS -> \"d\"\n else -> error(\"Unknown unit: $this\")\n}\n\n@SinceKotlin(\"1.5\")\ninternal fun durationUnitByShortName(shortName: String): DurationUnit = when (shortName) {\n \"ns\" -> DurationUnit.NANOSECONDS\n \"us\" -> DurationUnit.MICROSECONDS\n \"ms\" -> DurationUnit.MILLISECONDS\n \"s\" -> DurationUnit.SECONDS\n \"m\" -> DurationUnit.MINUTES\n \"h\" -> DurationUnit.HOURS\n \"d\" -> DurationUnit.DAYS\n else -> throw IllegalArgumentException(\"Unknown duration unit short name: $shortName\")\n}\n\n@SinceKotlin(\"1.5\")\ninternal fun durationUnitByIsoChar(isoChar: Char, isTimeComponent: Boolean): DurationUnit =\n when {\n !isTimeComponent -> {\n when (isoChar) {\n 'D' -> DurationUnit.DAYS\n else -> throw IllegalArgumentException(\"Invalid or unsupported duration ISO non-time unit: $isoChar\")\n }\n }\n else -> {\n when (isoChar) {\n 'H' -> DurationUnit.HOURS\n 'M' -> DurationUnit.MINUTES\n 'S' -> DurationUnit.SECONDS\n else -> throw IllegalArgumentException(\"Invalid duration ISO time unit: $isoChar\")\n }\n }\n}","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.time\n\nimport kotlin.annotation.AnnotationTarget.*\n\n/**\n * This annotation marks the experimental preview of the standard library API for measuring time and working with durations.\n *\n * > Note that this API is in a preview state and has a very high chance of being changed in the future.\n * Do not use it if you develop a library since your library will become binary incompatible\n * with the

future versions of the standard library.\n *\n * Any usage of a declaration annotated with `@ExperimentalTime` must be accepted either by\n * annotating that usage with the [OptIn] annotation, e.g. `@OptIn(ExperimentalTime::class)`,\n * or by using the compiler argument `-opt-in=kotlin.time.ExperimentalTime`.\n */\n@RequiresOptIn(level = RequiresOptIn.Level.ERROR)\n@MustBeDocumented\n@Retention(AnnotationRetention.BINARY)\n@Target(\n CLASS,\n ANNOTATION_CLASS,\n PROPERTY,\n FIELD,\n LOCAL_VARIABLE,\n VALUE_PARAMETER,\n CONSTRUCTOR,\n FUNCTION,\n PROPERTY_GETTER,\n PROPERTY_SETTER,\n TYPEALIAS\n)\n@SinceKotlin(\"1.3\")\npublic annotation class ExperimentalTime\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.time\n\n/**\n * A source of time for measuring time intervals.\n *\n * The only operation provided by the time source is [markNow]. It returns a [TimeMark], which can be used to query the elapsed time later.\n *\n * @see [measureTime]\n * @see [measureTimedValue]\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\npublic interface TimeSource {\n /**\n * Marks a point in time on this time source.\n *\n * The returned [TimeMark] instance encapsulates the captured time point and allows querying\n * the duration of time interval [elapsed][TimeMark.elapsedNow] from that point.\n */\n public fun markNow(): TimeMark\n\n /**\n * The most precise time source available in the platform.\n *\n * This time source returns its readings from a source of monotonic time when it is available in a target platform,\n * and resorts to a non-monotonic time source otherwise.\n */\n public object Monotonic : TimeSource by MonotonicTimeSource {\n override fun toString(): String = MonotonicTimeSource.toString()\n }\n\n\n public companion object {\n\n }\n}\n\n/**\n * Represents a time point notched on a particular [TimeSource]. Remains bound to the time source it was taken from\n * and allows querying for the duration of time elapsed from that point (see the function [elapsedNow]).\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\npublic abstract class TimeMark {\n /**\n * Returns the amount of time passed from this mark measured with the time source from which this mark was taken.\n *\n * Note that the value returned by this function can change on subsequent invocations.\n */\n public abstract fun elapsedNow(): Duration\n\n /**\n * Returns a time mark on the same time source that is ahead of this time mark by the specified [duration].\n *\n * The returned time mark is more _late_ when the [duration] is positive, and more _early_ when the [duration] is negative.\n */\n public open operator fun plus(duration: Duration): TimeMark = AdjustedTimeMark(this, duration)\n\n /**\n * Returns a time mark on the same time source that is behind this time mark by the specified [duration].\n *\n * The returned time mark is more _early_ when the [duration] is positive, and more _late_ when the [duration] is negative.\n */\n public open operator fun minus(duration: Duration): TimeMark = plus(-duration)\n\n\n /**\n * Returns true if this time mark has passed according to the time source from which this mark was taken.\n *\n * Note that the value returned by this function can change on subsequent invocations.\n * If the time source is monotonic, it can change only from `false` to `true`, namely, when the time mark becomes behind the current point of the time source.\n */\n public fun hasPassedNow(): Boolean = !elapsedNow().isNegative()\n\n /**\n * Returns false if this time mark has not passed according to the time source from which this mark was taken.\n *\n * Note that the value returned by this function can change on subsequent invocations.\n * If the time source is monotonic, it can change only from `true` to `false`, namely, when the time mark becomes behind the current point of the time source.\n */\n public fun hasNotPassedNow(): Boolean = elapsedNow().isNegative()\n}\n\n\n\n@ExperimentalTime\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Deprecated(\n \"Subtracting one TimeMark from another is not a well defined operation because these time marks could have been obtained from the different time sources.\",\n level = DeprecationLevel.ERROR\n)\n@Suppress(\"UNUSED_PARAMETER\")\npublic inline operator fun TimeMark.minus(other: TimeMark): Duration = throw Error(\"Operation is disallowed.\")\n\n@ExperimentalTime\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\n@Deprecated(\n \"Comparing one TimeMark to another is not a well defined operation because these time marks could have been obtained from the different time sources.\",\n level =

DeprecationLevel.ERROR\n)\n@Suppress(\"UNUSED_PARAMETER\")\npublic inline operator fun TimeMark.compareTo(other: TimeMark): Int = throw Error(\"Operation is disallowed.\")\n\n\n@ExperimentalTime\nprivate class AdjustedTimeMark(val mark: TimeMark, val adjustment: Duration) : TimeMark() {\n   override fun elapsedNow(): Duration = mark.elapsedNow() - adjustment\n\n   override fun plus(duration: Duration): TimeMark = AdjustedTimeMark(mark, adjustment + duration)\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.time\n\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\ninternal expect object MonotonicTimeSource : TimeSource\n\n/**\n * An abstract class used to implement time sources that return their readings as [Long] values in the specified [unit].\n *\n * @property unit The unit in which this time source's readings are expressed.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\npublic abstract class AbstractLongTimeSource(protected val unit: DurationUnit) : TimeSource {\n   /**\n   * This protected method should be overridden to return the current reading of the time source expressed as a [Long] number\n   * in the unit specified by the [unit] property.\n   */\n   protected abstract fun read(): Long\n\n   private class LongTimeMark(private val startedAt: Long, private val timeSource: AbstractLongTimeSource, private val offset: Duration) : TimeMark() {\n   override fun elapsedNow(): Duration = (timeSource.read() - startedAt).toDuration(timeSource.unit) - offset\n   override fun plus(duration: Duration): TimeMark = LongTimeMark(startedAt, timeSource, offset + duration)\n   }\n\n   override fun markNow(): TimeMark = LongTimeMark(read(), this, Duration.ZERO)\n}\n\n/**\n * An abstract class used to implement time sources that return their readings as [Double] values in the specified [unit].\n *\n * @property unit The unit in which this time source's readings are expressed.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\npublic abstract class AbstractDoubleTimeSource(protected val unit: DurationUnit) : TimeSource {\n   /**\n   * This protected method should be overridden to return the current reading of the time source expressed as a [Double] number\n   * in the unit specified by the [unit] property.\n   */\n   protected abstract fun read(): Double\n\n   private class DoubleTimeMark(private val startedAt: Double, private val timeSource: AbstractDoubleTimeSource, private val offset: Duration) : TimeMark() {\n   override fun elapsedNow(): Duration = (timeSource.read() - startedAt).toDuration(timeSource.unit) - offset\n   override fun plus(duration: Duration): TimeMark = DoubleTimeMark(startedAt, timeSource, offset + duration)\n   }\n\n   override fun markNow(): TimeMark = DoubleTimeMark(read(), this, Duration.ZERO)\n}\n\n/**\n * A time source that has programmatically updatable readings. It is useful as a predictable source of time in tests.\n *\n * The current reading value can be advanced by the specified duration amount with the operator [plusAssign]:\n *\n * ```\n * val timeSource = TestTimeSource()\n * timeSource += 10.seconds\n * ```\n *\n * Implementation note: the current reading value is stored as a [Long] number of nanoseconds,\n * thus it's capable to represent a time range of approximately \u00b1292 years.\n * Should the reading value overflow as the result of [plusAssign] operation, an [IllegalStateException] is thrown.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\npublic class TestTimeSource : AbstractLongTimeSource(unit = DurationUnit.NANOSECONDS) {\n   private var reading: Long = 0L\n\n   override fun read(): Long = reading\n\n   /**\n   * Advances the current reading value of this time source by the specified [duration].\n   *\n   * [duration] value is rounded down towards zero when converting it to a [Long] number of nanoseconds.\n   * For example, if the duration being added is `0.6.nanoseconds`, the reading doesn't advance because\n   * the duration value is rounded to zero nanoseconds.\n   *\n   * @throws IllegalStateException when the reading value overflows as the result of this operation.\n   */\n   public operator fun plusAssign(duration: Duration) {\n   val longDelta = duration.toLong(unit)\n   reading = if (longDelta != Long.MIN_VALUE && longDelta != Long.MAX_VALUE) {\n   // when delta fits in long, add it as long\n   val newReading = reading + longDelta\n   if (reading xor longDelta >= 0 && reading xor newReading < 0) overflow(duration)\n   newReading\n   } else {\n   val delta = duration.toDouble(unit)\n   // when delta is greater than long, add it as double\n   val newReading = reading + delta\n   if (newReading > Long.MAX_VALUE || newReading < Long.MIN_VALUE) overflow(duration)\n   newReading.toLong()\n   }\n   }\n\n   private fun overflow(duration: Duration) {\n   throw IllegalStateException(\"TestTimeSource will overflow if its reading ${reading}ns is advanced by $duration.\")\n   }\n}\n","/*\n * Copyright 2010-2020 JetBrains

s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.time\n\nimport kotlin.contracts.*\n\n/**\n * Executes the given function [block] and returns the duration of elapsed time interval.\n *\n * The elapsed time is measured with [TimeSource.Monotonic].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\npublic inline fun measureTime(block: () -> Unit): Duration {\n contract {\n callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n }\n return TimeSource.Monotonic.measureTime(block)\n}\n\n\n/**\n * Executes the given function [block] and returns the duration of elapsed time interval.\n *\n * The elapsed time is measured with the specified `this` [TimeSource] instance.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\npublic inline fun TimeSource.measureTime(block: () -> Unit): Duration {\n contract {\n callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n }\n\n val mark = markNow()\n block()\n return mark.elapsedNow()\n}\n\n\n/**\n * Data class representing a result of executing an action, along with the duration of elapsed time interval.\n *\n * @property value the result of the action.\n * @property duration the time elapsed to execute the action.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\npublic data class TimedValue<T>(val value: T, val duration: Duration)\n\n/**\n * Executes the given function [block] and returns an instance of [TimedValue] class, containing both\n * the result of the function execution and the duration of elapsed time interval.\n *\n * The elapsed time is measured with [TimeSource.Monotonic].\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\npublic inline fun <T> measureTimedValue(block: () -> T): TimedValue<T> {\n contract {\n callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n }\n\n return TimeSource.Monotonic.measureTimedValue(block)\n}\n\n/**\n * Executes the given [block] and returns an instance of [TimedValue] class, containing both\n * the result of function execution and the duration of elapsed time interval.\n *\n * The elapsed time is measured with the specified `this` [TimeSource] instance.\n */\n@SinceKotlin(\"1.3\")\n@ExperimentalTime\npublic inline fun <T> TimeSource.measureTimedValue(block: () -> T): TimedValue<T> {\n contract {\n callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n }\n\n val mark = markNow()\n val result = block()\n return TimedValue(result, mark.elapsedNow())\n}\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\nimport kotlin.coroutines.*\nimport kotlin.coroutines.intrinsics.*\nimport kotlin.native.concurrent.SharedImmutable\n\n/**\n * Defines deep recursive function that keeps its stack on the heap,\n * which allows very deep recursive computations that do not use the actual call stack.\n * To initiate a call to this deep recursive function use its [invoke] function.\n * As a rule of thumb, it should be used if recursion goes deeper than a thousand calls.\n *\n * The [DeepRecursiveFunction] takes one parameter of type [T] and returns a result of type [R].\n * The [block] of code defines the body of a recursive function. In this block\n * [callRecursive][DeepRecursiveScope.callRecursive] function can be used to make a recursive call\n * to the declared function. Other instances of [DeepRecursiveFunction] can be called\n * in this scope with `callRecursive` extension, too.\n *\n * For example, take a look at the following recursive tree class and a deeply\n * recursive instance of this tree with 100K nodes:\n *\n * ```\n * class Tree(val left: Tree? = null, val right: Tree? = null)\n * val deepTree = generateSequence(Tree()) { Tree(it) }.take(100_000).last()\n * ```\n *\n * A regular recursive function can be defined to compute a depth of a tree:\n *\n * ```\n * fun depth(t: Tree?): Int =\n *   if (t == null) 0 else max(depth(t.left), depth(t.right)) + 1\n * println(depth(deepTree)) // StackOverflowError\n * ```\n *\n * If this `depth` function is called for a `deepTree` it produces [StackOverflowError] because of deep recursion.\n * However, the `depth` function can be rewritten using `DeepRecursiveFunction` in the following way, and then\n * it successfully computes [`depth(deepTree)`][DeepRecursiveFunction.invoke] expression:\n *\n * ```\n * val depth = DeepRecursiveFunction<Tree?, Int> { t ->\n *   if (t == null) 0 else max(callRecursive(t.left), callRecursive(t.right)) + 1\n * }\n * println(depth(deepTree)) // Ok\n * ```\n *\n * Deep recursive functions can also mutually call each other using a heap for the stack via\n * [callRecursive][DeepRecursiveScope.callRecursive] extension. For example, the\n * following pair of mutually recursive functions computes the number of tree nodes at even depth in the tree.\n *\n * ```\n * val mutualRecursion = object {\n *   val even:

```
DeepRecursiveFunction<Tree?, Int> = DeepRecursiveFunction { t ->\n *         if (t == null) 0 else odd.callRecursive(t.left) + odd.callRecursive(t.right) + 1\n *     }\n *     val odd: DeepRecursiveFunction<Tree?, Int> = DeepRecursiveFunction { t ->\n *         if (t == null) 0 else even.callRecursive(t.left) + even.callRecursive(t.right)\n *     }\n * }\n * ```\n *\n * @param [T] the function parameter type.\n * @param [R] the function result type.\n * @param block the function body.\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalStdlibApi\npublic class DeepRecursiveFunction<T, R>(\n    internal val block: suspend DeepRecursiveScope<T, R>.(T) -> R\n)\n\n/**\n * Initiates a call to this deep recursive function, forming a root of the call tree.\n *\n * This operator should not be used from inside of [DeepRecursiveScope] as it uses the call stack slot for\n * initial recursive invocation. From inside of [DeepRecursiveScope] use\n * [callRecursive][DeepRecursiveScope.callRecursive].\n */\n@SinceKotlin(\"1.4\")\n@ExperimentalStdlibApi\npublic operator fun <T, R> DeepRecursiveFunction<T, R>.invoke(value: T): R =\n    DeepRecursiveScopeImpl<T, R>(block, value).runCallLoop()\n\n/**\n * A scope class for [DeepRecursiveFunction] function declaration that defines [callRecursive] methods to\n * recursively call this function or another [DeepRecursiveFunction] putting the call activation frame on the heap.\n *\n * @param [T] function parameter type.\n * @param [R] function result type.\n */\n@RestrictsSuspension\n@SinceKotlin(\"1.4\")\n@ExperimentalStdlibApi\npublic sealed class DeepRecursiveScope<T, R> {\n    /**\n     * Makes recursive call to this [DeepRecursiveFunction] function putting the call activation frame on the heap,\n     * as opposed to the actual call stack that is used by a regular recursive call.\n     */\n    public abstract suspend fun callRecursive(value: T): R\n\n    /**\n     * Makes call to the specified [DeepRecursiveFunction] function putting the call activation frame on the heap,\n     * as opposed to the actual call stack that is used by a regular call.\n     */\n    public abstract suspend fun <U, S> DeepRecursiveFunction<U, S>.callRecursive(value: U): S\n\n    @Deprecated(\n        level = DeprecationLevel.ERROR,\n        message =\n            \"'invoke' should not be called from DeepRecursiveScope. \" +\n                \"Use 'callRecursive' to do recursion in the heap instead of the call stack.\",\n        replaceWith = ReplaceWith(\"this.callRecursive(value)\")\n    )\n    @Suppress(\"UNUSED_PARAMETER\")\n    public operator fun DeepRecursiveFunction<*, *>.invoke(value: Any?): Nothing =\n        throw UnsupportedOperationException(\"Should not be called from DeepRecursiveScope\")\n}\n\n// ================== Implementation ==================\n\n@ExperimentalStdlibApi\nprivate typealias DeepRecursiveFunctionBlock = suspend DeepRecursiveScope<*, *>.(Any?) -> Any?\n\n@SharedImmutable\nprivate val UNDEFINED_RESULT = Result.success(COROUTINE_SUSPENDED)\n\n@Suppress(\"UNCHECKED_CAST\")\n@ExperimentalStdlibApi\nprivate class DeepRecursiveScopeImpl<T, R>(\n    block: suspend DeepRecursiveScope<T, R>.(T) -> R,\n    value: T\n) : DeepRecursiveScope<T, R>(), Continuation<R> {\n    // Active function block\n    private var function: DeepRecursiveFunctionBlock = block as DeepRecursiveFunctionBlock\n\n    // Value to call function with\n    private var value: Any? = value\n\n    // Continuation of the current call\n    private var cont: Continuation<Any?>? = this as Continuation<Any?>\n\n    // Completion result (completion of the whole call stack)\n    private var result: Result<Any?> = UNDEFINED_RESULT\n\n    override val context: CoroutineContext\n        get() = EmptyCoroutineContext\n\n    override fun resumeWith(result: Result<R>) {\n        this.cont = null\n        this.result = result\n    }\n\n    override suspend fun callRecursive(value: T): R =\n        suspendCoroutineUninterceptedOrReturn { cont ->\n            // calling the same function that is currently active\n            this.cont = cont as Continuation<Any?>\n            this.value = value\n            COROUTINE_SUSPENDED\n        }\n\n    override suspend fun <U, S> DeepRecursiveFunction<U, S>.callRecursive(value: U): S =\n        suspendCoroutineUninterceptedOrReturn { cont ->\n            // calling another recursive function\n            val function = block as DeepRecursiveFunctionBlock\n            with(this@DeepRecursiveScopeImpl) {\n                val currentFunction = this.function\n                if (function !== currentFunction) {\n                    // calling a different function -- create a trampoline to restore function ref\n                    this.function = function\n                    this.cont = crossFunctionCompletion(currentFunction, cont as Continuation<Any?>)\n                } else {\n                    // calling the same function -- direct\n                    this.cont = cont as Continuation<Any?>\n                }\n                this.value = value\n            }\n            COROUTINE_SUSPENDED\n        }\n\n    private fun crossFunctionCompletion(\n        currentFunction:
```

DeepRecursiveFunctionBlock,\n cont: Continuation<Any?>\n): Continuation<Any?> = Continuation(EmptyCoroutineContext) {\n this.function = currentFunction\n // When going back from a trampoline we cannot just call cont.resume (stack usage!)\n // We delegate the cont.resumeWith(it) call to runCallLoop\n this.cont = cont\n this.result = it\n }\n\n @Suppress(\"UNCHECKED_CAST\")\n fun runCallLoop(): R {\n while (true) {\n // Note: cont is set to null in DeepRecursiveScopeImpl.resumeWith when the whole computation completes\n val result = this.result\n val cont = this.cont\n ?: return (result as Result<R>).getOrThrow() // done -- final result\n // The order of comparison is important here for that case of rogue class with broken equals\n if (UNDEFINED_RESULT == result) {\n // call \"function\" with \"value\" using \"cont\" as completion\n val r = try {\n // This is block.startCoroutine(this, value, cont)\n function.startCoroutineUninterceptedOrReturn(this, value, cont)\n } catch (e: Throwable) {\n cont.resumeWithException(e)\n continue\n }\n // If the function returns without suspension -- calls its continuation immediately\n if (r !== COROUTINE_SUSPENDED)\n cont.resume(r as R)\n } else {\n // we returned from a crossFunctionCompletion trampoline -- call resume here\n this.result = UNDEFINED_RESULT // reset result back\n cont.resumeWith(result)\n }\n }\n }\n}\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\n@file:kotlin.jvm.JvmName(\"NumbersKt\")\n@file:kotlin.jvm.JvmMultifileClass\npackage kotlin\n\nimport kotlin.math.sign\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity. */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Byte.floorDiv(other: Byte): Int = \n this.toInt().floorDiv(other.toInt())\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Byte.mod(other: Byte): Byte = \n this.toInt().mod(other.toInt()).toByte()\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity. */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Byte.floorDiv(other: Short): Int = \n this.toInt().floorDiv(other.toInt())\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Byte.mod(other: Short): Short = \n this.toInt().mod(other.toInt()).toShort()\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity. */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Byte.floorDiv(other: Int): Int = \n this.toInt().floorDiv(other)\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Byte.mod(other: Int): Int = \n this.toInt().mod(other)\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity. */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Byte.floorDiv(other: Long): Long = \n this.toLong().floorDiv(other)\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Byte.mod(other: Long): Long = \n this.toLong().mod(other)\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity. */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Short.floorDiv(other: Byte): Int = \n this.toInt().floorDiv(other.toInt())\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Short.mod(other: Byte): Byte = \n this.toInt().mod(other.toInt()).toByte()\n\n/** Divides this value by the other

value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Short.floorDiv(other: Short): Int = \n this.toInt().floorDiv(other.toInt())\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Short.mod(other: Short): Short = \n   this.toInt().mod(other.toInt()).toShort()\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Short.floorDiv(other: Int): Int = \n this.toInt().floorDiv(other)\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Short.mod(other: Int): Int = \n   this.toInt().mod(other)\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Short.floorDiv(other: Long): Long = \n this.toLong().floorDiv(other)\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Short.mod(other: Long): Long = \n   this.toLong().mod(other)\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Int.floorDiv(other: Byte): Int = \n this.floorDiv(other.toInt())\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Int.mod(other: Byte): Byte = \n   this.mod(other.toInt()).toByte()\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Int.floorDiv(other: Short): Int = \n this.floorDiv(other.toInt())\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Int.mod(other: Short): Short = \n   this.mod(other.toInt()).toShort()\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Int.floorDiv(other: Int): Int {\n   var q = this / other\n   if (this xor other < 0 && q * other != this) q-- \n   return q\n}\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Int.mod(other: Int): Int {\n   val r = this % other\n   return r + (other and (((r xor other) and (r or -r)) shr 31))\n}\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Int.floorDiv(other: Long): Long = \n this.toLong().floorDiv(other)\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Int.mod(other: Long): Long = \n   this.toLong().mod(other)\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Long.floorDiv(other: Byte): Long = \n this.floorDiv(other.toLong())\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun

Long.mod(other: Byte): Byte = \n   this.mod(other.toLong()).toByte()\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Long.floorDiv(other: Short): Long = \n   this.floorDiv(other.toLong())\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Long.mod(other: Short): Short = \n   this.mod(other.toLong()).toShort()\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Long.floorDiv(other: Int): Long = \n   this.floorDiv(other.toLong())\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Long.mod(other: Int): Int = \n   this.mod(other.toLong()).toInt()\n\n/** Divides this value by the other value, flooring the result to an integer that is closer to negative infinity.
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Long.floorDiv(other: Long): Long {\n   var q = this / other\n   if (this xor other < 0 && q * other != this) q-- \n   return q\n}\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Long.mod(other: Long): Long {\n   val r = this % other\n   return r + (other and (((r xor other) and (r or -r)) shr 63))\n}\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n * \n * If the result cannot be represented exactly, it is rounded to the nearest representable number. In this case the absolute value of the result can be less than or _equal to_ the absolute value of the divisor.\n
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Float.mod(other: Float): Float {\n   val r = this % other\n   return if (r != 0.0.toFloat() && r.sign != other.sign) r + other else r\n}\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n * \n * If the result cannot be represented exactly, it is rounded to the nearest representable number. In this case the absolute value of the result can be less than or _equal to_ the absolute value of the divisor.\n
*/\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Float.mod(other: Double): Double = \n   this.toDouble().mod(other)\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n * \n * If the result cannot be represented exactly, it is rounded to the nearest representable number. In this case the absolute value of the result can be less than or _equal to_ the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Double.mod(other: Float): Double = \n   this.mod(other.toDouble())\n\n/**\n * Calculates the remainder of flooring division of this value by the other value.\n * \n * The result is either zero or has the same sign as the _divisor_ and has the absolute value less than the absolute value of the divisor.\n * \n * If the result cannot be represented exactly, it is rounded to the nearest representable number. In this case the absolute value of the result can be less than or _equal to_ the absolute value of the divisor.\n */\n@SinceKotlin(\"1.5\")\n@kotlin.internal.InlineOnly\npublic inline fun Double.mod(other: Double): Double {\n   val r = this % other\n   return if (r != 0.0 && r.sign != other.sign) r + other else r\n}\n\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\nimport kotlin.internal.InlineOnly\n\n\n/**\n * Returns a hash code value for the object or zero if the object is `null`.\n *\n * @see Any.hashCode\n */\n@SinceKotlin(\"1.3\")\n@InlineOnly\npublic inline fun Any?.hashCode(): Int = this?.hashCode() ?: 0\n","/*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be

found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\n/**\n * Represents a version of the Kotlin standard library.\n *\n * [major], [minor] and [patch] are integer components of a version,\n * they must be non-negative and not greater than 255 ([MAX_COMPONENT_VALUE]).\n *\n * @constructor Creates a version from all three components.\n */\n@SinceKotlin(\"1.1\")\npublic class KotlinVersion(val major: Int, val minor: Int, val patch: Int) : Comparable<KotlinVersion> {\n /**\n * Creates a version from [major] and [minor] components, leaving [patch] component zero.\n */\n public constructor(major: Int, minor: Int) : this(major, minor, 0)\n\n private val version = versionOf(major, minor, patch)\n\n private fun versionOf(major: Int, minor: Int, patch: Int): Int {\n require(major in 0..MAX_COMPONENT_VALUE && minor in 0..MAX_COMPONENT_VALUE && patch in 0..MAX_COMPONENT_VALUE) {\n \"Version components are out of range: $major.$minor.$patch\"\n }\n return major.shl(16) + minor.shl(8) + patch\n }\n\n /**\n * Returns the string representation of this version\n */\n override fun toString(): String = \"$major.$minor.$patch\"\n\n override fun equals(other: Any?): Boolean {\n if (this === other) return true\n val otherVersion = (other as? KotlinVersion) ?: return false\n return this.version == otherVersion.version\n }\n\n override fun hashCode(): Int = version\n\n override fun compareTo(other: KotlinVersion): Int = version - other.version\n\n /**\n * Returns `true` if this version is not less than the version specified\n * with the provided [major] and [minor] components.\n */\n public fun isAtLeast(major: Int, minor: Int): Boolean = // this.version >= versionOf(major, minor, 0)\n this.major > major || (this.major == major &&\n this.minor >= minor)\n\n /**\n * Returns `true` if this version is not less than the version specified\n * with the provided [major], [minor] and [patch] components.\n */\n public fun isAtLeast(major: Int, minor: Int, patch: Int): Boolean = // this.version >= versionOf(major, minor, patch)\n this.major > major || (this.major == major &&\n (this.minor > minor || this.minor == minor &&\n this.patch >= patch))\n\n companion object {\n /**\n * Maximum value a version component can have, a constant value 255.\n */\n // NOTE: Must be placed before CURRENT because its initialization requires this field being initialized in JS\n public const val MAX_COMPONENT_VALUE = 255\n\n /**\n * Returns the current version of the Kotlin standard library.\n */\n @kotlin.jvm.JvmField\n public val CURRENT: KotlinVersion = KotlinVersionCurrentValue.get()\n }\n}\n\n// this class is ignored during classpath normalization when considering whether to recompile dependencies in Kotlin build\nprivate object KotlinVersionCurrentValue {\n @kotlin.jvm.JvmStatic\n fun get(): KotlinVersion = KotlinVersion(1, 6, 21) // value is written here automatically during build\n}","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmName(\"LateinitKt\")\n@file:Suppress(\"unused\")\n\npackage kotlin\n\nimport kotlin.internal.InlineOnly\nimport kotlin.internal.AccessibleLateinitPropertyLiteral\nimport kotlin.reflect.KProperty0\n\n/**\n * Returns `true` if this lateinit property has been assigned a value, and `false` otherwise.\n *\n * Cannot be used in an inline function, to avoid binary compatibility issues.\n */\n@SinceKotlin(\"1.2\")\n@InlineOnly\ninline val @receiver:AccessibleLateinitPropertyLiteral KProperty0<*>.isInitialized: Boolean\n get() = throw NotImplementedError(\"Implementation is intrinsic\")\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmName(\"LazyKt\")\n@file:kotlin.jvm.JvmMultifileClass\n\npackage kotlin\n\nimport kotlin.reflect.KProperty\n\n/**\n * Represents a value with lazy initialization.\n *\n * To create an instance of [Lazy] use the [lazy] function.\n */\npublic interface Lazy<out T> {\n /**\n * Gets the lazily initialized value of the current Lazy instance.\n * Once the value was initialized it must not change during the rest of lifetime of this Lazy instance.\n */\n public val value: T\n\n /**\n * Returns `true` if a value for this Lazy instance has been already initialized, and `false` otherwise.\n * Once this function has returned `true` it stays `true` for the rest of lifetime of this Lazy instance.\n */\n public fun isInitialized(): Boolean\n}\n\n/**\n * Creates a new instance of the [Lazy] that is already initialized with the specified [value].\n */\npublic fun <T> lazyOf(value: T): Lazy<T> = InitializedLazyImpl(value)\n\n/**\n * An extension to delegate a read-only property of type [T] to an instance of [Lazy].\n *\n * This extension allows to use instances of Lazy for property delegation:\n * `val property: String by

lazy { initializer }`\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> Lazy<T>.getValue(thisRef: Any?, property: KProperty<*>): T = value\n\n/**\n * Specifies how a [Lazy] instance synchronizes initialization among multiple threads.\n */\npublic enum class LazyThreadSafetyMode {\n\n /**\n * Locks are used to ensure that only a single thread can initialize the [Lazy] instance.\n */\n SYNCHRONIZED,\n\n /**\n * Initializer function can be called several times on concurrent access to uninitialized [Lazy] instance value,\n * but only the first returned value will be used as the value of [Lazy] instance.\n */\n PUBLICATION,\n\n /**\n * No locks are used to synchronize an access to the [Lazy] instance value; if the instance is accessed from multiple threads, its behavior is undefined.\n *\n * This mode should not be used unless the [Lazy] instance is guaranteed never to be initialized from more than one thread.\n */\n NONE,\n}\n\n\ninternal object UNINITIALIZED_VALUE\n\n// internal to be called from lazy in JS\ninternal class UnsafeLazyImpl<out T>(initializer: () -> T) : Lazy<T>, Serializable {\n private var initializer: (() -> T)? = initializer\n private var _value: Any? = UNINITIALIZED_VALUE\n\n override val value: T\n get() {\n if (_value === UNINITIALIZED_VALUE) {\n _value = initializer!!()\n initializer = null\n }\n @Suppress(\"UNCHECKED_CAST\")\n return _value as T\n }\n\n override fun isInitialized(): Boolean = _value !== UNINITIALIZED_VALUE\n\n override fun toString(): String = if (isInitialized()) value.toString() else \"Lazy value not initialized yet.\"\n\n private fun writeReplace(): Any = InitializedLazyImpl(value)\n}\n\ninternal class InitializedLazyImpl<out T>(override val value: T) : Lazy<T>, Serializable {\n\n override fun isInitialized(): Boolean = true\n\n override fun toString(): String = value.toString()\n\n}\n","/*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"NumbersKt\")\npackage kotlin\n\n/**\n * Counts the number of set bits in the binary representation of this [Int] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Int.countOneBits(): Int\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Int.countLeadingZeroBits(): Int\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Int.countTrailingZeroBits(): Int\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [Int] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Int.takeHighestOneBit(): Int\n\n/**\n * Returns a number having a single bit set in the position of the least significant set bit of this [Int] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Int.takeLowestOneBit(): Int\n\n/**\n * Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n *\n * Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally\n * `number.rotateLeft(n) == number.rotateLeft(n % 32)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Int.rotateLeft(bitCount: Int): Int\n\n\n/**\n * Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n *\n * Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally\n * `number.rotateRight(n) ==

number.rotateRight(n % 32)`\n
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Int.rotateRight(bitCount: Int): Int\n\n\n/**\n * Counts the number of set bits in the binary representation of this [Long] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Long.countOneBits(): Int\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Long] number.\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Long.countLeadingZeroBits(): Int\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Long] number.\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Long.countTrailingZeroBits(): Int\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [Long] number,\n * or zero, if this number is zero.\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Long.takeHighestOneBit(): Long\n\n/**\n * Returns a number having a single bit set in the position of the least significant set bit of this [Long] number,\n * or zero, if this number is zero.\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Long.takeLowestOneBit(): Long\n\n/**\n * Rotates the binary representation of this [Long] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n *\n * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n * `number.rotateLeft(n) == number.rotateLeft(n % 64)`\n
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Long.rotateLeft(bitCount: Int): Long\n\n/**\n * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n *\n * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n * `number.rotateRight(n) == number.rotateRight(n % 64)`\n
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic expect fun Long.rotateRight(bitCount: Int): Long\n\n/**\n * Counts the number of set bits in the binary representation of this [Byte] number.\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun Byte.countOneBits(): Int = (toInt() and 0xFF).countOneBits()\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Byte] number.\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun Byte.countLeadingZeroBits(): Int = (toInt() and 0xFF).countLeadingZeroBits() - (Int.SIZE_BITS - Byte.SIZE_BITS)\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Byte] number.\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun Byte.countTrailingZeroBits(): Int = (toInt() or 0x100).countTrailingZeroBits()\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [Byte] number,\n * or zero, if this number is zero.\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun Byte.takeHighestOneBit(): Byte = (toInt() and 0xFF).takeHighestOneBit().toByte()\n\n/**\n * Returns a number having a single bit set in the position of the least significant set bit of this [Byte] number,\n * or zero, if this number is zero.\n
*/\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npubli

c inline fun Byte.takeLowestOneBit(): Byte = toInt().takeLowestOneBit().toByte()\n\n/**\n * Rotates the binary representation of this [Byte] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n *\n * Rotating by a multiple of [Byte.SIZE_BITS] (8) returns the same number, or more generally\n * `number.rotateLeft(n) == number.rotateLeft(n % 8)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun Byte.rotateLeft(bitCount: Int): Byte =\n   (toInt().shl(bitCount and 7) or (toInt() and 0xFF).ushr(8 - (bitCount and 7))).toByte()\n\n/**\n * Rotates the binary representation of this [Byte] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n *\n * Rotating by a multiple of [Byte.SIZE_BITS] (8) returns the same number, or more generally\n * `number.rotateRight(n) == number.rotateRight(n % 8)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun Byte.rotateRight(bitCount: Int): Byte =\n   (toInt().shl(8 - (bitCount and 7)) or (toInt() and 0xFF).ushr(bitCount and 7)).toByte()\n\n/**\n * Counts the number of set bits in the binary representation of this [Short] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun Short.countOneBits(): Int = (toInt() and 0xFFFF).countOneBits()\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Short] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun Short.countLeadingZeroBits(): Int =\n   (toInt() and 0xFFFF).countLeadingZeroBits() - (Int.SIZE_BITS - Short.SIZE_BITS)\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Short] number.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun Short.countTrailingZeroBits(): Int = (toInt() or 0x10000).countTrailingZeroBits()\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [Short] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun Short.takeHighestOneBit(): Short = (toInt() and 0xFFFF).takeHighestOneBit().toShort()\n\n/**\n * Returns a number having a single bit set in the position of the least significant set bit of this [Short] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun Short.takeLowestOneBit(): Short = toInt().takeLowestOneBit().toShort()\n\n/**\n * Rotates the binary representation of this [Short] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n *\n * Rotating by a multiple of [Short.SIZE_BITS] (16) returns the same number, or more generally\n * `number.rotateLeft(n) == number.rotateLeft(n % 16)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun Short.rotateLeft(bitCount: Int): Short =\n   (toInt().shl(bitCount and 15) or (toInt() and 0xFFFF).ushr(16 - (bitCount and 15))).toShort()\n\n/**\n * Rotates the binary representation of this [Short] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n *\n * Rotating by a multiple of [Short.SIZE_BITS] (16) returns the same number, or more generally\n * `number.rotateRight(n) == number.rotateRight(n % 16)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic fun

Short.rotateRight(bitCount: Int): Short =\n   (toInt().shl(16 - (bitCount and 15)) or (toInt() and 0xFFFF).ushr(bitCount and 15)).toShort()\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\nimport kotlin.internal.RequireKotlin\nimport kotlin.internal.RequireKotlinVersionKind\n\n@kotlin.internal.InlineOnly\n@SinceKotlin(\"1.2\")\n@Suppress(\"INVISIBLE_MEMBER\", \"INVISIBLE_REFERENCE\")\n@RequireKotlin(\"1.2.30\", level = DeprecationLevel.HIDDEN, versionKind = RequireKotlinVersionKind.COMPILER_VERSION)\npublic inline fun <R> suspend(noinline block: suspend () -> R): suspend () -> R = block\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmName(\"TuplesKt\")\n\npackage kotlin\n\n\n/**\n * Represents a generic pair of two values.\n *\n * There is no meaning attached to values in this class, it can be used for any purpose.\n * Pair exhibits value semantics, i.e. two pairs are equal if both components are equal.\n *\n * An example of decomposing it into values:\n * @sample samples.misc.Tuples.pairDestructuring\n *\n * @param A type of the first value.\n * @param B type of the second value.\n * @property first First value.\n * @property second Second value.\n * @constructor Creates a new instance of Pair.\n */\npublic data class Pair<out A, out B>(\n   public val first: A,\n   public val second: B\n) : Serializable {\n\n   /**\n   * Returns string representation of the [Pair] including its [first] and [second] values.\n   */\n   public override fun toString(): String = \"($first, $second)\"\n}\n\n/**\n * Creates a tuple of type [Pair] from this and [that].\n *\n * This can be useful for creating [Map] literals with less noise, for example:\n * @sample samples.collections.Maps.Instantiation.mapFromPairs\n */\npublic infix fun <A, B> A.to(that: B): Pair<A, B> = Pair(this, that)\n\n/**\n * Converts this pair into a list.\n * @sample samples.misc.Tuples.pairToList\n */\npublic fun <T> Pair<T, T>.toList(): List<T> = listOf(first, second)\n\n/**\n * Represents a triad of values\n *\n * There is no meaning attached to values in this class, it can be used for any purpose.\n * Triple exhibits value semantics, i.e. two triples are equal if all three components are equal.\n * An example of decomposing it into values:\n * @sample samples.misc.Tuples.tripleDestructuring\n *\n * @param A type of the first value.\n * @param B type of the second value.\n * @param C type of the third value.\n * @property first First value.\n * @property second Second value.\n * @property third Third value.\n */\npublic data class Triple<out A, out B, out C>(\n   public val first: A,\n   public val second: B,\n   public val third: C\n) : Serializable {\n\n   /**\n   * Returns string representation of the [Triple] including its [first], [second] and [third] values.\n   */\n   public override fun toString(): String = \"($first, $second, $third)\"\n}\n\n/**\n * Converts this triple into a list.\n * @sample samples.misc.Tuples.tripleToList\n */\npublic fun <T> Triple<T, T, T>.toList(): List<T> = listOf(first, second, third)\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin.ranges\n\n\n\nimport kotlin.internal.*\n\n/**\n * A range of values of type `UInt`.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic class UIntRange(start: UInt, endInclusive: UInt) : UIntProgression(start, endInclusive, 1), ClosedRange<UInt> {\n   override val start: UInt get() = first\n   override val endInclusive: UInt get() = last\n\n   override fun contains(value: UInt): Boolean = first <= value && value <= last\n\n   /** \n   * Checks if the range is empty.\n   \n   * The range is empty if its start value is greater than the end value.\n   */\n   override fun isEmpty(): Boolean = first > last\n\n   override fun equals(other: Any?): Boolean =\n   other is UIntRange && (isEmpty() && other.isEmpty() ||\n   first == other.first && last == other.last)\n\n   override fun hashCode(): Int =\n   if (isEmpty()) -1 else (31 * first.toInt() + last.toInt())\n\n   override fun toString(): String = \"$first..$last\"\n\n   companion object {\n   /** An empty range of values of type UInt. */\n   public val EMPTY: UIntRange = UIntRange(UInt.MAX_VALUE, UInt.MIN_VALUE)\n   }\n}\n\n/**\n * A progression of values of type `UInt`.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic open class UIntProgression\ninternal constructor(\n   start: UInt,\n   endInclusive: UInt,\n   step: Int\n) : Iterable<UInt> {\n   init {\n   if (step == 0.toInt()) throw kotlin.IllegalArgumentException(\"Step must be non-zero.\")\n   if (step

== Int.MIN_VALUE) throw kotlin.IllegalArgumentException(\"Step must be greater than Int.MIN_VALUE to avoid overflow on negation.\")\n   }\n\n   /**\n   * The first element in the progression.\n   */\n   public val first: UInt = start\n\n   /**\n   * The last element in the progression.\n   */\n   public val last: UInt = getProgressionLastElement(start, endInclusive, step)\n\n   /**\n   * The step of the progression.\n   */\n   public val step: Int = step\n\n   final override fun iterator(): Iterator<UInt> = UIntProgressionIterator(first, last, step)\n\n   /** \n   * Checks if the progression is empty.\n   \n   * Progression with a positive step is empty if its first element is greater than the last element.\n   * Progression with a negative step is empty if its first element is less than the last element.\n   */\n   public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last\n\n   override fun equals(other: Any?): Boolean =\n   other is UIntProgression && (isEmpty() && other.isEmpty() ||\n   first == other.first && last == other.last && step == other.step)\n\n   override fun hashCode(): Int =\n   if (isEmpty()) -1 else (31 * (31 * first.toInt() + last.toInt()) + step.toInt())\n\n   override fun toString(): String = if (step > 0) \"$first..$last step $step\" else \"$first downTo $last step ${-step}\"\n\n   companion object {\n   /**\n   * Creates UIntProgression within the specified bounds of a closed range.\n\n   * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].\n   * In order to go backwards the [step] must be negative.\n   *\n   * [step] must be greater than `Int.MIN_VALUE` and not equal to zero.\n   */\n   public fun fromClosedRange(rangeStart: UInt, rangeEnd: UInt, step: Int): UIntProgression = UIntProgression(rangeStart, rangeEnd, step)\n   }\n}\n\n\n/**\n * An iterator over a progression of values of type `UInt`.\n * @property step the number by which the value is incremented on each step.\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"DEPRECATION_ERROR\")\nprivate class UIntProgressionIterator(first: UInt, last: UInt, step: Int) : UIntIterator() {\n   private val finalElement = last\n   private var hasNext: Boolean = if (step > 0) first <= last else first >= last\n   private val step = step.toUInt() // use 2-complement math for negative steps\n   private var next = if (hasNext) first else finalElement\n\n   override fun hasNext(): Boolean = hasNext\n\n   override fun nextUInt(): UInt {\n   val value = next\n   if (value == finalElement) {\n   if (!hasNext) throw kotlin.NoSuchElementException()\n   hasNext = false\n   } else {\n   next += step\n   }\n   return value\n   }\n}\n\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin.collections\n\n/** An iterator over a sequence of values of type `UByte`. */\n@Deprecated(\"This class is not going to be stabilized and is to be removed soon.\", level = DeprecationLevel.ERROR)\n@SinceKotlin(\"1.3\")\npublic abstract class UByteIterator : Iterator<UByte> {\n   final override fun next() = nextUByte()\n\n   /** Returns the next value in the sequence without boxing. */\n   public abstract fun nextUByte(): UByte\n}\n\n/** An iterator over a sequence of values of type `UShort`. */\n@Deprecated(\"This class is not going to be stabilized and is to be removed soon.\", level = DeprecationLevel.ERROR)\n@SinceKotlin(\"1.3\")\npublic abstract class UShortIterator : Iterator<UShort> {\n   final override fun next() = nextUShort()\n\n   /** Returns the next value in the sequence without boxing. */\n   public abstract fun nextUShort(): UShort\n}\n\n/** An iterator over a sequence of values of type `UInt`. */\n@Deprecated(\"This class is not going to be stabilized and is to be removed soon.\", level = DeprecationLevel.ERROR)\n@SinceKotlin(\"1.3\")\npublic abstract class UIntIterator : Iterator<UInt> {\n   final override fun next() = nextUInt()\n\n   /** Returns the next value in the sequence without boxing. */\n   public abstract fun nextUInt(): UInt\n}\n\n/** An iterator over a sequence of values of type `ULong`. */\n@Deprecated(\"This class is not going to be stabilized and is to be removed soon.\", level = DeprecationLevel.ERROR)\n@SinceKotlin(\"1.3\")\npublic abstract class ULongIterator : Iterator<ULong> {\n   final override fun next() = nextULong()\n\n   /** Returns the next value in the sequence without boxing. */\n   public abstract fun nextULong(): ULong\n}\n\n","/*\n * Copyright 2010-2022 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n// Auto-generated file. DO NOT EDIT!\n\npackage kotlin.ranges\n\n\n\nimport kotlin.internal.*\n\n/**\n * A range of values of type `ULong`.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic class ULongRange(start: ULong, endInclusive: ULong) : ULongProgression(start, endInclusive, 1),

ClosedRange<ULong> {\n override val start: ULong get() = first\n override val endInclusive: ULong get() = last\n\n override fun contains(value: ULong): Boolean = first <= value && value <= last\n\n /** \n * Checks if the range is empty.\n \n * The range is empty if its start value is greater than the end value.\n */\n override fun isEmpty(): Boolean = first > last\n\n override fun equals(other: Any?): Boolean =\n other is ULongRange && (isEmpty() && other.isEmpty() ||\n first == other.first && last == other.last)\n\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt())\n\n override fun toString(): String = \"$first..$last\"\n\n companion object {\n /** An empty range of values of type ULong. */\n public val EMPTY: ULongRange = ULongRange(ULong.MAX_VALUE, ULong.MIN_VALUE)\n }\n}\n\n/**\n * A progression of values of type `ULong`.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic open class ULongProgression\ninternal constructor(\n start: ULong,\n endInclusive: ULong,\n step: Long\n) : Iterable<ULong> {\n init {\n if (step == 0.toLong()) throw kotlin.IllegalArgumentException(\"Step must be non-zero.\")\n if (step == Long.MIN_VALUE) throw kotlin.IllegalArgumentException(\"Step must be greater than Long.MIN_VALUE to avoid overflow on negation.\")\n }\n\n /**\n * The first element in the progression.\n */\n public val first: ULong = start\n\n /**\n * The last element in the progression.\n */\n public val last: ULong = getProgressionLastElement(start, endInclusive, step)\n\n /**\n * The step of the progression.\n */\n public val step: Long = step\n\n final override fun iterator(): Iterator<ULong> = ULongProgressionIterator(first, last, step)\n\n /** \n * Checks if the progression is empty.\n \n * Progression with a positive step is empty if its first element is greater than the last element.\n * Progression with a negative step is empty if its first element is less than the last element.\n */\n public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last\n\n override fun equals(other: Any?): Boolean =\n other is ULongProgression && (isEmpty() && other.isEmpty() ||\n first == other.first && last == other.last && step == other.step)\n\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt()) + (step xor (step ushr 32)).toInt())\n\n override fun toString(): String = if (step > 0) \"$first..$last step $step\" else \"$first downTo $last step ${-step}\"\n\n companion object {\n /**\n * Creates ULongProgression within the specified bounds of a closed range.\n\n * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].\n * In order to go backwards the [step] must be negative.\n *\n * [step] must be greater than `Long.MIN_VALUE` and not equal to zero.\n */\n public fun fromClosedRange(rangeStart: ULong, rangeEnd: ULong, step: Long): ULongProgression = ULongProgression(rangeStart, rangeEnd, step)\n }\n}\n\n\n/**\n * An iterator over a progression of values of type `ULong`.\n * @property step the number by which the value is incremented on each step.\n */\n@SinceKotlin(\"1.3\")\n@Suppress(\"DEPRECATION_ERROR\")\nprivate class ULongProgressionIterator(first: ULong, last: ULong, step: Long) : ULongIterator() {\n private val finalElement = last\n private var hasNext: Boolean = if (step > 0) first <= last else first >= last\n private val step = step.toULong() // use 2-complement math for negative steps\n private var next = if (hasNext) first else finalElement\n\n override fun hasNext(): Boolean = hasNext\n\n override fun nextULong(): ULong {\n val value = next\n if (value == finalElement) {\n if (!hasNext) throw kotlin.NoSuchElementException()\n hasNext = false\n } else {\n next += step\n }\n return value\n }\n}\n\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.math\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun min(a: UInt, b: UInt): UInt {\n return minOf(a, b)\n}\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun min(a: ULong, b: ULong): ULong {\n return minOf(a, b)\n}\n\n/**\n * Returns the greater of two values.\n

*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun max(a: UInt, b: UInt): UInt {\n   return maxOf(a, b)\n}\n\n/**\n * Returns the greater of two values.\n
/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun max(a: ULong, b: ULong): ULong {\n   return maxOf(a, b)\n}","/\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*/\n\n@file:kotlin.jvm.JvmName(\"UNumbersKt\")\npackage kotlin\n\n/**\n * Counts the number of set bits in the binary representation of this [UInt] number.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UInt.countOneBits(): Int = toInt().countOneBits()\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UInt] number.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UInt.countLeadingZeroBits(): Int = toInt().countLeadingZeroBits()\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UInt] number.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UInt.countTrailingZeroBits(): Int = toInt().countTrailingZeroBits()\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [UInt] number,\n * or zero, if this number is zero.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UInt.takeHighestOneBit(): UInt = toInt().takeHighestOneBit().toUInt()\n\n/**\n * Returns a number having a single bit set in the position of the least significant set bit of this [UInt] number,\n * or zero, if this number is zero.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UInt.takeLowestOneBit(): UInt = toInt().takeLowestOneBit().toUInt()\n\n/**\n * Rotates the binary representation of this [UInt] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n *\n * Rotating by a multiple of [UInt.SIZE_BITS] (32) returns the same number, or more generally\n * `number.rotateLeft(n) == number.rotateLeft(n % 32)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class, ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UInt.rotateLeft(bitCount: Int): UInt = toInt().rotateLeft(bitCount).toUInt()\n\n/**\n * Rotates the binary representation of this [UInt] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n *\n * Rotating by a multiple of [UInt.SIZE_BITS] (32) returns the same number, or more generally\n * `number.rotateRight(n) == number.rotateRight(n % 32)`\n
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class, ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UInt.rotateRight(bitCount: Int): UInt = toInt().rotateRight(bitCount).toUInt()\n\n/**\n * Counts the number of set bits in the binary representation of this [ULong] number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULong.countOneBits(): Int = toLong().countOneBits()\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [ULong] number.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class,

ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULong.countLeadingZeroBits(): Int = toLong().countLeadingZeroBits()\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [ULong] number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULong.countTrailingZeroBits(): Int = toLong().countTrailingZeroBits()\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [ULong] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULong.takeHighestOneBit(): ULong = toLong().takeHighestOneBit().toULong()\n\n/**\n * Returns a number having a single bit set in the position of the least significant set bit of this [ULong] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULong.takeLowestOneBit(): ULong = toLong().takeLowestOneBit().toULong()\n\n/**\n * Rotates the binary representation of this [ULong] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n *\n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally\n * `number.rotateLeft(n) == number.rotateLeft(n % 64)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class, ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULong.rotateLeft(bitCount: Int): ULong = toLong().rotateLeft(bitCount).toULong()\n\n/**\n * Rotates the binary representation of this [ULong] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n *\n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally\n * `number.rotateRight(n) == number.rotateRight(n % 64)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class, ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun ULong.rotateRight(bitCount: Int): ULong = toLong().rotateRight(bitCount).toULong()\n\n/**\n * Counts the number of set bits in the binary representation of this [UByte] number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByte.countOneBits(): Int = toUInt().countOneBits()\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UByte] number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByte.countLeadingZeroBits(): Int = toByte().countLeadingZeroBits()\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UByte] number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByte.countTrailingZeroBits(): Int = toByte().countTrailingZeroBits()\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [UByte] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByte.takeHighestOneBit(): UByte = toInt().takeHighestOneBit().toUByte()\n\n/**\n * Returns a number having a single bit set in the position of the least significant set bit of this [UByte] number,\n * or zero, if this number is zero.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class,

ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByte.takeLowestOneBit(): UByte = toInt().takeLowestOneBit().toUByte()\n\n\n/**\n * Rotates the binary representation of this [UByte] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n *\n * Rotating by a multiple of [UByte.SIZE_BITS] (8) returns the same number, or more generally\n * `number.rotateLeft(n) == number.rotateLeft(n % 8)`\n
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class, ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByte.rotateLeft(bitCount: Int): UByte = toByte().rotateLeft(bitCount).toUByte()\n\n/**\n * Rotates the binary representation of this [UByte] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n *\n * Rotating by a multiple of [UByte.SIZE_BITS] (8) returns the same number, or more generally\n * `number.rotateRight(n) == number.rotateRight(n % 8)`\n
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class, ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UByte.rotateRight(bitCount: Int): UByte = toByte().rotateRight(bitCount).toUByte()\n\n/**\n * Counts the number of set bits in the binary representation of this [UShort] number.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShort.countOneBits(): Int = toUInt().countOneBits()\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UShort] number.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShort.countLeadingZeroBits(): Int = toShort().countLeadingZeroBits()\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UShort] number.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShort.countTrailingZeroBits(): Int = toShort().countTrailingZeroBits()\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [UShort] number,\n * or zero, if this number is zero.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShort.takeHighestOneBit(): UShort = toInt().takeHighestOneBit().toUShort()\n\n/**\n * Returns a number having a single bit set in the position of the least significant set bit of this [UShort] number,\n * or zero, if this number is zero.\n
*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class, ExperimentalStdlibApi::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShort.takeLowestOneBit(): UShort = toInt().takeLowestOneBit().toUShort()\n\n\n/**\n * Rotates the binary representation of this [UShort] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * `number.rotateLeft(-n) == number.rotateRight(n)`\n *\n * Rotating by a multiple of [UShort.SIZE_BITS] (16) returns the same number, or more generally\n * `number.rotateLeft(n) == number.rotateLeft(n % 16)`\n
*/\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class, ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShort.rotateLeft(bitCount: Int): UShort = toShort().rotateLeft(bitCount).toUShort()\n\n/**\n * Rotates the binary representation of this [UShort] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a

negative bit count is the same as rotating it left by the negated bit count:\n * `number.rotateRight(-n) == number.rotateLeft(n)`\n *\n * Rotating by a multiple of [UShort.SIZE_BITS] (16) returns the same number, or more generally\n * `number.rotateRight(n) == number.rotateRight(n % 16)`\n */\n@SinceKotlin(\"1.6\")\n@WasExperimental(ExperimentalStdlibApi::class, ExperimentalUnsignedTypes::class)\n@kotlin.internal.InlineOnly\npublic inline fun UShort.rotateRight(bitCount: Int): UShort = toShort().rotateRight(bitCount).toUShort()\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.internal\n\n// (a - b) mod c\nprivate fun differenceModulo(a: UInt, b: UInt, c: UInt): UInt {\n   val ac = a % c\n   val bc = b % c\n   return if (ac >= bc) ac - bc else ac - bc + c\n}\n\nprivate fun differenceModulo(a: ULong, b: ULong, c: ULong): ULong {\n   val ac = a % c\n   val bc = b % c\n   return if (ac >= bc) ac - bc else ac - bc + c\n}\n\n/**\n * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range\n * from [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative\n * [step].\n *\n * No validation on passed parameters is performed. The given parameters should satisfy the condition:\n *\n * - either `step > 0` and `start <= end`,\n * - or `step < 0` and `start >= end`.\n *\n * @param start first element of the progression\n * @param end ending bound for the progression\n * @param step increment, or difference of successive elements in the progression\n * @return the final element of the progression\n * @suppress\n */\n@PublishedApi\n@SinceKotlin(\"1.3\")\ninternal fun getProgressionLastElement(start: UInt, end: UInt, step: Int): UInt = when {\n   step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step.toUInt())\n   step < 0 -> if (start <= end) end else end + differenceModulo(start, end, (-step).toUInt())\n   else -> throw kotlin.IllegalArgumentException(\"Step is zero.\")\n}\n\n/**\n * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range\n * from [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative\n * [step].\n *\n * No validation on passed parameters is performed. The given parameters should satisfy the condition:\n *\n * - either `step > 0` and `start <= end`,\n * - or `step < 0` and `start >= end`.\n *\n * @param start first element of the progression\n * @param end ending bound for the progression\n * @param step increment, or difference of successive elements in the progression\n * @return the final element of the progression\n * @suppress\n */\n@PublishedApi\n@SinceKotlin(\"1.3\")\ninternal fun getProgressionLastElement(start: ULong, end: ULong, step: Long): ULong = when {\n   step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step.toULong())\n   step < 0 -> if (start <= end) end else end + differenceModulo(start, end, (-step).toULong())\n   else -> throw kotlin.IllegalArgumentException(\"Step is zero.\")\n}\n","/*\n * Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmName(\"UStringsKt\") // string representation of unsigned numbers\n\npackage kotlin.text\n\n/**\n * Returns a string representation of this [Byte] value in the specified [radix].\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n//@kotlin.internal.InlineOnly\npublic /*inline*/ fun UByte.toString(radix: Int): String = this.toInt().toString(radix)\n\n/**\n * Returns a string representation of this [Short] value in the specified [radix].\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n//@kotlin.internal.InlineOnly\npublic /*inline*/ fun UShort.toString(radix: Int): String = this.toInt().toString(radix)\n\n\n/**\n * Returns a string representation of this [Int] value in the specified [radix].\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\n//@kotlin.internal.InlineOnly\npublic /*inline*/ fun UInt.toString(radix: Int): String = this.toLong().toString(radix)\n\n/**\n * Returns a string representation of this [Long] value in the specified [radix].\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\n

*/\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun ULong.toString(radix: Int): String = ulongToString(this.toLong(), checkRadix(radix))\n\n\n/**\n * Parses the string as a signed [UByte] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUByte(): UByte = toUByteOrNull() ?: numberFormatError(this)\n\n/**\n * Parses the string as a signed [UByte] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUByte(radix: Int): UByte = toUByteOrNull(radix) ?: numberFormatError(this)\n\n\n/**\n * Parses the string as a [UShort] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUShort(): UShort = toUShortOrNull() ?: numberFormatError(this)\n\n/**\n * Parses the string as a [UShort] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUShort(radix: Int): UShort = toUShortOrNull(radix) ?: numberFormatError(this)\n\n/**\n * Parses the string as an [UInt] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUInt(): UInt = toUIntOrNull() ?: numberFormatError(this)\n\n/**\n * Parses the string as an [UInt] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUInt(radix: Int): UInt = toUIntOrNull(radix) ?: numberFormatError(this)\n\n/**\n * Parses the string as a [ULong] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toULong(): ULong = toULongOrNull() ?: numberFormatError(this)\n\n/**\n * Parses the string as a [ULong] number and returns the result.\n * @throws NumberFormatException if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toULong(radix: Int): ULong = toULongOrNull(radix) ?: numberFormatError(this)\n\n\n\n\n\n/**\n * Parses the string as an [UByte] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUByteOrNull(): UByte? = toUByteOrNull(radix = 10)\n\n/**\n * Parses the string as an [UByte] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUByteOrNull(radix: Int): UByte? {\n   val int = this.toUIntOrNull(radix) ?: return null\n   if (int > UByte.MAX_VALUE) return null\n   return int.toUByte()\n}\n\n/**\n * Parses the string as an [UShort] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUShortOrNull(): UShort? = toUShortOrNull(radix = 10)\n\n/**\n * Parses the string as an [UShort] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun

```
String.toUShortOrNull(radix: Int): UShort? {\n    val int = this.toUIntOrNull(radix) ?: return null\n    if (int > UShort.MAX_VALUE) return null\n    return int.toUShort()\n}\n\n/**\n * Parses the string as an [UInt] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUIntOrNull(): UInt? = toUIntOrNull(radix = 10)\n\n/**\n * Parses the string as an [UInt] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toUIntOrNull(radix: Int): UInt? {\n    checkRadix(radix)\n\n    val length = this.length\n    if (length == 0) return null\n\n    val limit: UInt = UInt.MAX_VALUE\n    val start: Int\n\n    val firstChar = this[0]\n    if (firstChar < '0') {\n        if (length == 1 || firstChar != '+') return null\n        start = 1\n    } else {\n        start = 0\n    }\n\n    val limitForMaxRadix = 119304647u  //  limit / 36\n\n    var limitBeforeMul = limitForMaxRadix\n    val uradix = radix.toUInt()\n    var result = 0u\n    for (i in start until length) {\n        val digit = digitOf(this[i], radix)\n\n        if (digit < 0) return null\n        if (result > limitBeforeMul) {\n            if (limitBeforeMul == limitForMaxRadix) {\n                limitBeforeMul = limit / uradix\n\n                if (result > limitBeforeMul) {\n                    return null\n                }\n            } else {\n                return null\n            }\n        }\n\n        result *= uradix\n\n        val beforeAdding = result\n        result += digit.toUInt()\n        if (result < beforeAdding) return null // overflow has happened\n    }\n\n    return result\n}\n\n/**\n * Parses the string as an [ULong] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toULongOrNull(): ULong? = toULongOrNull(radix = 10)\n\n/**\n * Parses the string as an [ULong] number and returns the result\n * or `null` if the string is not a valid representation of a number.\n *\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n */\n@SinceKotlin(\"1.5\")\n@WasExperimental(ExperimentalUnsignedTypes::class)\npublic fun String.toULongOrNull(radix: Int): ULong? {\n    checkRadix(radix)\n\n    val length = this.length\n    if (length == 0) return null\n\n    val limit: ULong = ULong.MAX_VALUE\n    val start: Int\n\n    val firstChar = this[0]\n    if (firstChar < '0') {\n        if (length == 1 || firstChar != '+') return null\n        start = 1\n    } else {\n        start = 0\n    }\n\n\n    val limitForMaxRadix = 512409557603043100uL  //  limit / 36\n\n    var limitBeforeMul = limitForMaxRadix\n    val uradix = radix.toULong()\n    var result = 0uL\n    for (i in start until length) {\n        val digit = digitOf(this[i], radix)\n\n        if (digit < 0) return null\n        if (result > limitBeforeMul) {\n            if (limitBeforeMul == limitForMaxRadix) {\n                limitBeforeMul = limit / uradix\n\n                if (result > limitBeforeMul) {\n                    return null\n                }\n            } else {\n                return null\n            }\n        }\n\n        result *= uradix\n\n        val beforeAdding = result\n        result += digit.toUInt()\n        if (result < beforeAdding) return null // overflow has happened\n    }\n\n    return result\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:Suppress(\"INVISIBLE_REFERENCE\", \"INVISIBLE_MEMBER\")\npackage kotlin\n\nimport kotlin.annotation.AnnotationTarget.*\nimport kotlin.internal.RequireKotlin\nimport kotlin.internal.RequireKotlinVersionKind\n\n/**\n * Marks the API that is dependent on the experimental unsigned types, including those types themselves.\n *\n * Usages of such API will be reported as warnings unless an explicit opt-in with\n * the [OptIn] annotation, e.g. `@OptIn(ExperimentalUnsignedTypes::class)`,\n * or with the `-opt-in=kotlin.ExperimentalUnsignedTypes` compiler option is given.\n *\n * It's recommended to propagate the experimental status to the API that depends on unsigned types by annotating it with this annotation.\n */\n@RequiresOptIn(level = RequiresOptIn.Level.WARNING)\n@MustBeDocumented\n@Target(CLASS, ANNOTATION_CLASS, PROPERTY, FIELD, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, TYPEALIAS)\n@Retention(AnnotationRetention.BINARY)\n@RequireKotlin(\"1.2.50\", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)\npublic annotation class ExperimentalUnsignedTypes\n","/*\n
```

```
* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"MathKt\")\n\n\npackage kotlin.math\n\n\n\n// constants, can't use them from nativeMath as they are not constants there\n\n/** Ratio of the circumference of a circle to its diameter, approximately 3.14159. */\n@SinceKotlin(\"1.2\")\npublic const val PI: Double = 3.141592653589793\n/** Base of the natural logarithms, approximately 2.71828. */\n@SinceKotlin(\"1.2\")\npublic const val E: Double = 2.718281828459045\n\n// region ================ Double Math ====================================\n\n/** Computes the sine of the angle [x] given in radians.\n *\n *  Special cases:\n *   - `sin(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun sin(x: Double): Double\n\n/** Computes the cosine of the angle [x] given in radians.\n *\n *  Special cases:\n *   - `cos(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun cos(x: Double): Double\n\n/** Computes the tangent of the angle [x] given in radians.\n *\n *  Special cases:\n *   - `tan(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun tan(x: Double): Double\n\n/**\n * Computes the arc sine of the value [x];\n * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.\n *\n * Special cases:\n *    - `asin(x)` is `NaN`, when `abs(x) > 1` or x is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun asin(x: Double): Double\n\n/**\n * Computes the arc cosine of the value [x];\n * the returned value is an angle in the range from `0.0` to `PI` radians.\n *\n * Special cases:\n *    - `acos(x)` is `NaN`, when `abs(x) > 1` or x is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun acos(x: Double): Double\n\n/**\n * Computes the arc tangent of the value [x];\n * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.\n *\n * Special cases:\n *   - `atan(NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun atan(x: Double): Double\n\n/**\n * Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond\n * to the rectangular coordinates `(x, y)` by computing the arc tangent of the value [y] / [x];\n * the returned value is an angle in the range from `-PI` to `PI` radians.\n *\n * Special cases:\n *   - `atan2(0.0, 0.0)` is `0.0`\n *   - `atan2(0.0, x)` is  `0.0` for `x > 0` and `PI` for `x < 0`\n *   - `atan2(-0.0, x)` is `-0.0` for 'x > 0` and `-PI` for `x < 0`\n *   - `atan2(y, +Inf)` is `0.0` for `0 < y < +Inf` and `-0.0` for '-Inf < y < 0`\n *   - `atan2(y, -Inf)` is `PI` for `0 < y < +Inf` and `-PI` for `-Inf < y < 0`\n *   - `atan2(y, 0.0)` is `PI/2` for `y > 0` and `-PI/2` for `y < 0`\n *   - `atan2(+Inf, x)` is `PI/2` for finite `x`y\n *   - `atan2(-Inf, x)` is `-PI/2` for finite `x`\n *   - `atan2(NaN, x)` and `atan2(y, NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun atan2(y: Double, x: Double): Double\n\n/**\n * Computes the hyperbolic sine of the value [x].\n *\n * Special cases:\n *   - `sinh(NaN)` is `NaN`\n *   - `sinh(+Inf)` is `+Inf`\n *   - `sinh(-Inf)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun sinh(x: Double): Double\n\n/**\n * Computes the hyperbolic cosine of the value [x].\n *\n * Special cases:\n *   - `cosh(NaN)` is `NaN`\n *   - `cosh(+Inf|-Inf)` is `+Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun cosh(x: Double): Double\n\n/**\n * Computes the hyperbolic tangent of the value [x].\n *\n * Special cases:\n *   - `tanh(NaN)` is `NaN`\n *   - `tanh(+Inf)` is `1.0`\n *   - `tanh(-Inf)` is `-1.0`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun tanh(x: Double): Double\n\n/**\n * Computes the inverse hyperbolic sine of the value [x].\n *\n * The returned value is `y` such that `sinh(y) == x`.\n *\n * Special cases:\n *   - `asinh(NaN)` is `NaN`\n *   - `asinh(+Inf)` is `+Inf`\n *   - `asinh(-Inf)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun asinh(x: Double): Double\n\n/**\n * Computes the inverse hyperbolic cosine of the value [x].\n *\n * The returned value is positive `y` such that `cosh(y) == x`.\n *\n * Special cases:\n *   - `acosh(NaN)` is `NaN`\n *   - `acosh(x)` is `NaN` when `x < 1`\n *   - `acosh(+Inf)` is `+Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun acosh(x: Double): Double\n\n/**\n * Computes the inverse hyperbolic tangent of the value [x].\n *\n * The returned value is `y` such that `tanh(y) == x`.\n *\n * Special cases:\n *   - `tanh(NaN)` is `NaN`\n *   - `tanh(x)` is `NaN` when `x > 1` or `x < -1`\n *   - `tanh(1.0)` is `+Inf`\n *   - `tanh(-1.0)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun atanh(x: Double): Double\n\n/**\n * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.\n *\n * Special cases:\n *   - returns `+Inf` if any of arguments is infinite\n *   - returns `NaN` if any of arguments is `NaN` and the other is not infinite\n */\n@SinceKotlin(\"1.2\")\npublic expect fun hypot(x: Double, y: Double): Double\n\n/**\n * Computes the positive square root of the value [x].\n *\n * Special cases:\n *   - `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun sqrt(x: Double): Double\n\n/**\n * Computes Euler's number `e`
```

raised to the power of the value [x].\n *\n * Special cases:\n * - `exp(NaN)` is `NaN`\n * - `exp(+Inf)` is `+Inf`\n * - `exp(-Inf)` is `0.0`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun exp(x: Double): Double\n\n/**\n * Computes `exp(x) - 1`.\n *\n * This function can be implemented to produce more precise result for [x] near zero.\n *\n * Special cases:\n * - `expm1(NaN)` is `NaN`\n * - `expm1(+Inf)` is `+Inf`\n * - `expm1(-Inf)` is `-1.0`\n *\n * @see [exp] function.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun expm1(x: Double): Double\n\n/**\n * Computes the logarithm of the value [x] to the given [base].\n *\n * Special cases:\n * - `log(x, b)` is `NaN` if either `x` or `b` are `NaN`\n * - `log(x, b)` is `NaN` when `x < 0` or `b <= 0` or `b == 1.0`\n * - `log(+Inf, +Inf)` is `NaN`\n * - `log(+Inf, b)` is `+Inf` for `b > 1` and `-Inf` for `b < 1`\n * - `log(0.0, b)` is `-Inf` for `b > 1` and `+Inf` for `b > 1`\n *\n * See also logarithm functions for common fixed bases: [ln], [log10] and [log2].\n */\n@SinceKotlin(\"1.2\")\npublic expect fun log(x: Double, base: Double): Double\n\n/**\n * Computes the natural logarithm (base `E`) of the value [x].\n *\n * Special cases:\n * - `ln(NaN)` is `NaN`\n * - `ln(x)` is `NaN` when `x < 0.0`\n * - `ln(+Inf)` is `+Inf`\n * - `ln(0.0)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun ln(x: Double): Double\n\n/**\n * Computes the common logarithm (base 10) of the value [x].\n *\n * @see [ln] function for special cases.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun log10(x: Double): Double\n\n/**\n * Computes the binary logarithm (base 2) of the value [x].\n *\n * @see [ln] function for special cases.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun log2(x: Double): Double\n\n/**\n * Computes `ln(x + 1)`.\n *\n * This function can be implemented to produce more precise result for [x] near zero.\n *\n * Special cases:\n * - `ln1p(NaN)` is `NaN`\n * - `ln1p(x)` is `NaN` where `x < -1.0`\n * - `ln1p(-1.0)` is `-Inf`\n * - `ln1p(+Inf)` is `+Inf`\n *\n * @see [ln] function\n * @see [expm1] function\n */\n@SinceKotlin(\"1.2\")\npublic expect fun ln1p(x: Double): Double\n\n/**\n * Rounds the given value [x] to an integer towards positive infinity.\n\n * @return the smallest double value that is greater than or equal to the given value [x] and is a mathematical integer.\n *\n * Special cases:\n * - `ceil(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun ceil(x: Double): Double\n\n/**\n * Rounds the given value [x] to an integer towards negative infinity.\n\n * @return the largest double value that is smaller than or equal to the given value [x] and is a mathematical integer.\n *\n * Special cases:\n * - `floor(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun floor(x: Double): Double\n\n/**\n * Rounds the given value [x] to an integer towards zero.\n *\n * @return the value [x] having its fractional part truncated.\n *\n * Special cases:\n * - `truncate(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun truncate(x: Double): Double\n\n/**\n * Rounds the given value [x] towards the closest integer with ties rounded towards even integer.\n *\n * Special cases:\n * - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun round(x: Double): Double\n\n/**\n * Returns the absolute value of the given value [x].\n *\n * Special cases:\n * - `abs(NaN)` is `NaN`\n *\n * @see absoluteValue extension property for [Double]\n */\n@SinceKotlin(\"1.2\")\npublic expect fun abs(x: Double): Double\n\n/**\n * Returns the sign of the given value [x]:\n * - `-1.0` if the value is negative,\n * - zero if the value is zero,\n * - `1.0` if the value is positive\n *\n * Special case:\n * - `sign(NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun sign(x: Double): Double\n\n\n/**\n * Returns the smaller of two values.\n *\n * If either value is `NaN`, then the result is `NaN`.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun min(a: Double, b: Double): Double\n\n/**\n * Returns the greater of two values.\n *\n * If either value is `NaN`, then the result is `NaN`.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun max(a: Double, b: Double): Double\n\n// extensions\n\n/**\n * Raises this value to the power [x].\n *\n * Special cases:\n * - `b.pow(0.0)` is `1.0`\n * - `b.pow(1.0) == b`\n * - `b.pow(NaN)` is `NaN`\n * - `NaN.pow(x)` is `NaN` for `x != 0.0`\n * - `b.pow(Inf)` is `NaN` for `abs(b) == 1.0`\n * - `b.pow(x)` is `NaN` for `b < 0` and `x` is finite and not an integer\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Double.pow(x: Double): Double\n\n/**\n * Raises this value to the integer power [n].\n *\n * See the other overload of [pow] for details.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Double.pow(n: Int): Double\n\n/**\n * Returns the absolute value of this value.\n *\n * Special cases:\n * - `NaN.absoluteValue` is `NaN`\n *\n * @see abs function\n */\n@SinceKotlin(\"1.2\")\npublic expect val Double.absoluteValue: Double\n\n/**\n * Returns the sign of this value:\n * - `-1.0` if the value is negative,\n * - zero if the value is

zero,\n * - `1.0` if the value is positive\n *\n * Special case:\n * - `NaN.sign` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect val Double.sign: Double\n\n/**\n * Returns this value with the sign bit same as of the [sign] value.\n *\n * If [sign] is `NaN` the sign of the result is undefined.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Double.withSign(sign: Double): Double\n\n/**\n * Returns this value with the sign bit same as of the [sign] value.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Double.withSign(sign: Int): Double\n\n/**\n * Returns the ulp (unit in the last place) of this value.\n *\n * An ulp is a positive distance between this value and the next nearest [Double] value larger in magnitude.\n *\n * Special Cases:\n * - `NaN.ulp` is `NaN`\n * - `x.ulp` is `+Inf` when `x` is `+Inf` or `-Inf`\n * - `0.0.ulp` is `Double.MIN_VALUE`\n */\n@SinceKotlin(\"1.2\")\npublic expect val Double.ulp: Double\n\n/**\n * Returns the [Double] value nearest to this value in direction of positive infinity.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Double.nextUp(): Double\n\n/**\n * Returns the [Double] value nearest to this value in direction of negative infinity.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Double.nextDown(): Double\n\n/**\n * Returns the [Double] value nearest to this value in direction from this value towards the value [to].\n *\n * Special cases:\n * - `x.nextTowards(y)` is `NaN` if either `x` or `y` are `NaN`\n * - `x.nextTowards(x) == x`\n *\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Double.nextTowards(to: Double): Double\n\n/**\n * Rounds this [Double] value to the nearest integer and converts the result to [Int].\n * Ties are rounded towards positive infinity.\n *\n * Special cases:\n * - `x.roundToInt() == Int.MAX_VALUE` when `x > Int.MAX_VALUE`\n * - `x.roundToInt() == Int.MIN_VALUE` when `x < Int.MIN_VALUE`\n *\n * @throws IllegalArgumentException when this value is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Double.roundToInt(): Int\n\n/**\n * Rounds this [Double] value to the nearest integer and converts the result to [Long].\n * Ties are rounded towards positive infinity.\n *\n * Special cases:\n * - `x.roundToLong() == Long.MAX_VALUE` when `x > Long.MAX_VALUE`\n * - `x.roundToLong() == Long.MIN_VALUE` when `x < Long.MIN_VALUE`\n *\n * @throws IllegalArgumentException when this value is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Double.roundToLong(): Long\n\n// endregion\n\n\n\n// region ================ Float Math ======================================\n\n/** Computes the sine of the angle [x] given in radians.\n *\n * Special cases:\n * - `sin(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun sin(x: Float): Float\n\n/** Computes the cosine of the angle [x] given in radians.\n *\n * Special cases:\n * - `cos(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun cos(x: Float): Float\n\n/** Computes the tangent of the angle [x] given in radians.\n *\n * Special cases:\n * - `tan(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun tan(x: Float): Float\n\n/**\n * Computes the arc sine of the value [x];\n * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.\n *\n * Special cases:\n * - `asin(x)` is `NaN`, when `abs(x) > 1` or x is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun asin(x: Float): Float\n\n/**\n * Computes the arc cosine of the value [x];\n * the returned value is an angle in the range from `0.0` to `PI` radians.\n *\n * Special cases:\n * - `acos(x)` is `NaN`, when `abs(x) > 1` or x is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun acos(x: Float): Float\n\n/**\n * Computes the arc tangent of the value [x];\n * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.\n *\n * Special cases:\n * - `atan(NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun atan(x: Float): Float\n\n/**\n * Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond\n * to the rectangular coordinates `(x, y)` by computing the arc tangent of the value [y] / [x];\n * the returned value is an angle in the range from `-PI` to `PI` radians.\n *\n * Special cases:\n * - `atan2(0.0, 0.0)` is `0.0`\n * - `atan2(0.0, x)` is `0.0` for `x > 0` and `PI` for `x < 0`\n * - `atan2(-0.0, x)` is `-0.0` for 'x > 0` and `-PI` for `x < 0`\n *  - `atan2(y, +Inf)` is `0.0` for `0 < y < +Inf` and `-0.0` for '-Inf < y < 0`\n * - `atan2(y, -Inf)` is `PI` for `0 < y < +Inf` and `-PI` for `-Inf < y < 0`\n * - `atan2(y, 0.0)` is `PI/2` for `y > 0` and `-PI/2` for `y < 0`\n * - `atan2(+Inf, x)` is `PI/2` for finite `x`y\n * - `atan2(-Inf, x)` is `-PI/2` for finite `x`\n * - `atan2(NaN, x)` and `atan2(y, NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun atan2(y: Float, x: Float): Float\n\n/**\n * Computes the hyperbolic sine of the value [x].\n *\n * Special cases:\n * - `sinh(NaN)` is `NaN`\n * - `sinh(+Inf)` is `+Inf`\n * - `sinh(-Inf)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun sinh(x: Float): Float\n\n/**\n * Computes the hyperbolic cosine of the value [x].\n *\n * Special cases:\n * - `cosh(NaN)` is `NaN`\n * - `cosh(+Inf|-Inf)` is

`+Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun cosh(x: Float): Float\n\n/**\n * Computes the hyperbolic tangent of the value [x].\n *\n * Special cases:\n * - `tanh(NaN)` is `NaN`\n * - `tanh(+Inf)` is `1.0`\n * - `tanh(-Inf)` is `-1.0`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun tanh(x: Float): Float\n\n/**\n * Computes the inverse hyperbolic sine of the value [x].\n *\n * The returned value is `y` such that `sinh(y) == x`.\n *\n * Special cases:\n * - `asinh(NaN)` is `NaN`\n * - `asinh(+Inf)` is `+Inf`\n * - `asinh(-Inf)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun asinh(x: Float): Float\n\n/**\n * Computes the inverse hyperbolic cosine of the value [x].\n *\n * The returned value is positive `y` such that `cosh(y) == x`.\n *\n * Special cases:\n * - `acosh(NaN)` is `NaN`\n * - `acosh(x)` is `NaN` when `x < 1`\n * - `acosh(+Inf)` is `+Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun acosh(x: Float): Float\n\n/**\n * Computes the inverse hyperbolic tangent of the value [x].\n *\n * The returned value is `y` such that `tanh(y) == x`.\n *\n * Special cases:\n * - `tanh(NaN)` is `NaN`\n * - `tanh(x)` is `NaN` when `x > 1` or `x < -1`\n * - `tanh(1.0)` is `+Inf`\n * - `tanh(-1.0)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun atanh(x: Float): Float\n\n/**\n * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.\n *\n * Special cases:\n * - returns `+Inf` if any of arguments is infinite\n * - returns `NaN` if any of arguments is `NaN` and the other is not infinite\n */\n@SinceKotlin(\"1.2\")\npublic expect fun hypot(x: Float, y: Float): Float\n\n/**\n * Computes the positive square root of the value [x].\n *\n * Special cases:\n * - `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun sqrt(x: Float): Float\n\n/**\n * Computes Euler's number `e` raised to the power of the value [x].\n *\n * Special cases:\n * - `exp(NaN)` is `NaN`\n * - `exp(+Inf)` is `+Inf`\n * - `exp(-Inf)` is `0.0`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun exp(x: Float): Float\n\n/**\n * Computes `exp(x) - 1`.\n *\n * This function can be implemented to produce more precise result for [x] near zero.\n *\n * Special cases:\n * - `expm1(NaN)` is `NaN`\n * - `expm1(+Inf)` is `+Inf`\n * - `expm1(-Inf)` is `-1.0`\n *\n * @see [exp] function.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun expm1(x: Float): Float\n\n/**\n * Computes the logarithm of the value [x] to the given [base].\n *\n * Special cases:\n * - `log(x, b)` is `NaN` if either `x` or `b` are `NaN`\n * - `log(x, b)` is `NaN` when `x < 0` or `b <= 0` or `b == 1.0`\n * - `log(+Inf, +Inf)` is `NaN`\n * - `log(+Inf, b)` is `+Inf` for `b > 1` and `-Inf` for `b < 1`\n * - `log(0.0, b)` is `-Inf` for `b > 1` and `+Inf` for `b > 1`\n *\n * See also logarithm functions for common fixed bases: [ln], [log10] and [log2].\n */\n@SinceKotlin(\"1.2\")\npublic expect fun log(x: Float, base: Float): Float\n\n/**\n * Computes the natural logarithm (base `E`) of the value [x].\n *\n * Special cases:\n * - `ln(NaN)` is `NaN`\n * - `ln(x)` is `NaN` when `x < 0.0`\n * - `ln(+Inf)` is `+Inf`\n * - `ln(0.0)` is `-Inf`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun ln(x: Float): Float\n\n/**\n * Computes the common logarithm (base 10) of the value [x].\n *\n * @see [ln] function for special cases.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun log10(x: Float): Float\n\n/**\n * Computes the binary logarithm (base 2) of the value [x].\n *\n * @see [ln] function for special cases.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun log2(x: Float): Float\n\n/**\n * Computes `ln(a + 1)`.\n *\n * This function can be implemented to produce more precise result for [x] near zero.\n *\n * Special cases:\n * - `ln1p(NaN)` is `NaN`\n * - `ln1p(x)` is `NaN` where `x < -1.0`\n * - `ln1p(-1.0)` is `-Inf`\n * - `ln1p(+Inf)` is `+Inf`\n *\n * @see [ln] function\n * @see [expm1] function\n */\n@SinceKotlin(\"1.2\")\npublic expect fun ln1p(x: Float): Float\n\n/**\n * Rounds the given value [x] to an integer towards positive infinity.\n\n * @return the smallest Float value that is greater than or equal to the given value [x] and is a mathematical integer.\n *\n * Special cases:\n * - `ceil(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun ceil(x: Float): Float\n\n/**\n * Rounds the given value [x] to an integer towards negative infinity.\n\n * @return the largest Float value that is smaller than or equal to the given value [x] and is a mathematical integer.\n *\n * Special cases:\n * - `floor(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun floor(x: Float): Float\n\n/**\n * Rounds the given value [x] to an integer towards zero.\n *\n * @return the value [x] having its fractional part truncated.\n *\n * Special cases:\n * - `truncate(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun truncate(x: Float): Float\n\n/**\n * Rounds the given value [x] towards the closest integer with ties rounded towards even integer.\n *\n * Special cases:\n * - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n */\n@SinceKotlin(\"1.2\")\npublic expect

fun round(x: Float): Float\n\n\n/**\n * Returns the absolute value of the given value [x].\n *\n * Special cases:\n * - `abs(NaN)` is `NaN`\n *\n * @see absoluteValue extension property for [Float]\n */\n@SinceKotlin(\"1.2\")\npublic expect fun abs(x: Float): Float\n\n/**\n * Returns the sign of the given value [x]:\n * - `-1.0` if the value is negative,\n * - zero if the value is zero,\n * - `1.0` if the value is positive\n *\n * Special case:\n * - `sign(NaN)` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun sign(x: Float): Float\n\n\n\n/**\n * Returns the smaller of two values.\n *\n * If either value is `NaN`, then the result is `NaN`.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun min(a: Float, b: Float): Float\n\n/**\n * Returns the greater of two values.\n *\n * If either value is `NaN`, then the result is `NaN`.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun max(a: Float, b: Float): Float\n\n// extensions\n\n\n\n/**\n * Raises this value to the power [x].\n *\n * Special cases:\n * - `b.pow(0.0)` is `1.0`\n * - `b.pow(1.0) == b`\n * - `b.pow(NaN)` is `NaN`\n * - `NaN.pow(x)` is `NaN` for `x != 0.0`\n * - `b.pow(Inf)` is `NaN` for `abs(b) == 1.0`\n * - `b.pow(x)` is `NaN` for `b < 0` and `x` is finite and not an integer\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Float.pow(x: Float): Float\n\n/**\n * Raises this value to the integer power [n].\n *\n * See the other overload of [pow] for details.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Float.pow(n: Int): Float\n\n/**\n * Returns the absolute value of this value.\n *\n * Special cases:\n * - `NaN.absoluteValue` is `NaN`\n *\n * @see abs function\n */\n@SinceKotlin(\"1.2\")\npublic expect val Float.absoluteValue: Float\n\n/**\n * Returns the sign of this value:\n * - `-1.0` if the value is negative,\n * - zero if the value is zero,\n * - `1.0` if the value is positive\n *\n * Special case:\n * - `NaN.sign` is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect val Float.sign: Float\n\n/**\n * Returns this value with the sign bit same as of the [sign] value.\n *\n * If [sign] is `NaN` the sign of the result is undefined.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Float.withSign(sign: Float): Float\n\n/**\n * Returns this value with the sign bit same as of the [sign] value.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Float.withSign(sign: Int): Float\n\n\n\n/**\n * Rounds this [Float] value to the nearest integer and converts the result to [Int].\n * Ties are rounded towards positive infinity.\n *\n * Special cases:\n * - `x.roundToInt() == Int.MAX_VALUE` when `x > Int.MAX_VALUE`\n * - `x.roundToInt() == Int.MIN_VALUE` when `x < Int.MIN_VALUE`\n *\n * @throws IllegalArgumentException when this value is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Float.roundToInt(): Int\n\n/**\n * Rounds this [Float] value to the nearest integer and converts the result to [Long].\n * Ties are rounded towards positive infinity.\n *\n * Special cases:\n * - `x.roundToLong() == Long.MAX_VALUE` when `x > Long.MAX_VALUE`\n * - `x.roundToLong() == Long.MIN_VALUE` when `x < Long.MIN_VALUE`\n *\n * @throws IllegalArgumentException when this value is `NaN`\n */\n@SinceKotlin(\"1.2\")\npublic expect fun Float.roundToLong(): Long\n\n\n// endregion\n\n// region ================ Integer Math ==\n\n\n\n/**\n * Returns the absolute value of the given value [n].\n *\n * Special cases:\n * - `abs(Int.MIN_VALUE)` is `Int.MIN_VALUE` due to an overflow\n *\n * @see absoluteValue extension property for [Int]\n */\n@SinceKotlin(\"1.2\")\npublic expect fun abs(n: Int): Int\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun min(a: Int, b: Int): Int\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun max(a: Int, b: Int): Int\n\n/**\n * Returns the absolute value of this value.\n *\n * Special cases:\n * - `Int.MIN_VALUE.absoluteValue` is `Int.MIN_VALUE` due to an overflow\n *\n * @see abs function\n */\n@SinceKotlin(\"1.2\")\npublic expect val Int.absoluteValue: Int\n\n/**\n * Returns the sign of this value:\n * - `-1` if the value is negative,\n * - `0` if the value is zero,\n * - `1` if the value is positive\n */\n@SinceKotlin(\"1.2\")\npublic expect val Int.sign: Int\n\n\n\n\n/**\n * Returns the absolute value of the given value [n].\n *\n * Special cases:\n * - `abs(Long.MIN_VALUE)` is `Long.MIN_VALUE` due to an overflow\n *\n * @see absoluteValue extension property for [Long]\n */\n@SinceKotlin(\"1.2\")\npublic expect fun abs(n: Long): Long\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun min(a: Long, b: Long): Long\n\n/**\n * Returns the greater of two values.\n */\n@SinceKotlin(\"1.2\")\npublic expect fun max(a: Long, b: Long): Long\n\n/**\n * Returns the absolute value of this value.\n *\n * Special cases:\n * - `Long.MIN_VALUE.absoluteValue` is `Long.MIN_VALUE` due to an overflow\n *\n * @see abs function\n */\n@SinceKotlin(\"1.2\")\npublic expect val Long.absoluteValue: Long\n\n/**\n * Returns the sign of this value:\n * - `-1` if the value is negative,\n * - `0` if the value is zero,\n * - `1` if the value is positive\n

*/\n@SinceKotlin(\"1.2\")\npublic expect val Long.sign: Int\n\n\n//
endregion\n"],"names":[],"mappings":"AAWC,CAXA,yB;EACG,IAAI,OAAO,MAAO,KAAI,UAAW,IAAG,MAA
M,IAA1C,C;IACI,MAAM,CAAC,QAAD,EAAW,CAAC,SAAD,CAAX,EAAwB,OAAxB,C;SAEL,IAAI,OAAO,O
AAQ,KAAI,QAAvB,C;IACD,OAAO,CAAC,MAAM,QAAP,C;;IAGP,IAAI,OAAQ,GAAE,E;IACd,OAAO,CAAC
,IAAI,OAAL,C;;CAEd,CAAC,IAAD,EAAO,kB;EACJ,IAAI,IAAI,M;ECPZ,MAAM,eAAgB,GAAE,a;IACpB,OA
AoD,CAA5C,KAAK,QAAQ,CAAC,CAAD,CAAI,IAAG,CAAE,YAAW,SAAW,KAAG,CAAC,OAAQ,KAAI,c;G
;EAGxE,MAAM,YAAa,GAAE,a;IACjB,OAAO,CAAE,YAAW,SAAU,IAAG,CAAC,OAAQ,KAAI,c;G;EAGlD,M
AAM,aAAc,GAAE,a;IAClB,OAAO,CAAE,YAAW,U;G;EAGxB,MAAM,YAAa,GAAE,a;IACjB,OAAO,CAAE,Y
AAW,WAAY,IAAG,CAAC,OAAQ,KAAI,W;G;EAGpD,MAAM,WAAY,GAAE,a;IAChB,OAAO,CAAE,YAAW,
U;G;EAGxB,MAAM,aAAc,GAAE,a;IAClB,OAAO,CAAE,YAAW,Y;G;EAGxB,MAAM,cAAe,GAAE,a;IACnB,O
AAO,CAAE,YAAW,Y;G;EAGxB,MAAM,YAAa,GAAE,a;IACjB,OAAO,KAAK,QAAQ,CAAC,CAAD,CAAI,IA
AG,CAAC,OAAQ,KAAI,W;G;EAG5C,MAAM,QAAS,GAAE,a;IACb,OAAO,KAAK,QAAQ,CAAC,CAAD,CAA
I,IAAG,CAAC,CAAC,O;G;EAGjC,MAAM,WAAY,GAAE,a;IAChB,OAAO,KAAK,QAAQ,CAAC,CAAD,CAAI,
IAAG,WAAW,OAAO,CAAC,CAAD,C;G;EAGjD,MAAM,cAAe,GAAE,a;IACnB,IAAI,CAAE,KAAI,IAAV,C;M
AAgB,OAAO,M;IACvB,IAAI,WAAW,MAAM,YAAY,CAAC,CAAD,CAAI,GAAE,MAAM,aAAR,GAAwB,MA
AM,S;IACnE,OAAO,GAAI,GAAE,KAAK,UAAU,IAAI,KAAK,CAAC,CAAD,EAAI,a;MAAc,OAAO,QAAQ,CA
AC,CAAD,C;KAAjC,CAAwC,KAAK,CAAC,IAAD,CAAO,GAAE,G;G;EAG/F,MAAM,kBAAmB,GAAE,e;IACv
B,OAAO,MAAM,OAAO,YAAY,wBAAwB,CAAC,GAAD,C;G;EAG5D,MAAM,YAAa,GAAE,gB;IACjB,IAAI,C
AAE,KAAI,CAAV,C;MACI,OAAO,I;;IAEX,IAAI,CAAE,KAAI,IAAK,IAAG,CAAE,KAAI,IAAK,IAAG,CAAC,
MAAM,WAAW,CAAC,CAAD,CAAI,IAAG,CAAC,OAAQ,KAAI,CAAC,OAAvE,C;MACI,OAAO,K;;IAGX,KA
AK,IAAI,IAAI,CAAR,EAAW,IAAI,CAAC,OAArB,EAA8B,CAAE,GAAE,CAAlC,EAAqC,CAAC,EAAtC,C;MA
CI,IAAI,CAAC,MAAM,OAAO,CAAC,CAAC,CAAC,CAAD,CAAF,EAAO,CAAC,CAAC,CAAD,CAAR,CAAlB
,C;QACI,OAAO,K;;;IAGf,OAAO,I;G;EAGX,MAAM,gBAAiB,GAAE,gB;IACrB,OAAO,MAAM,OAAO,YAAY,s
BAAsB,CAAC,CAAD,EAAI,CAAJ,C;G;EAG1D,MAAM,cAAe,GAAE,e;IACnB,IAAI,GAAI,KAAI,IAAZ,C;MA
AkB,OAAO,C;IACzB,IAAI,SAAS,C;IACb,KAAK,IAAI,IAAI,CAAR,EAAW,IAAI,GAAG,OAAvB,EAAgC,CAA
E,GAAE,CAApC,EAAuC,CAAC,EAAxC,C;MACI,MAAO,GAAqB,CAAjB,EAAG,GAAE,MAAO,GAAE,CAAG
,IAAE,MAAM,SAAS,CAAC,GAAG,CAAC,CAAD,CAAJ,CAAU,GAAE,C;;IAE7D,OAAO,M;G;EAGX,MAAM,
kBAAmB,GAAE,e;IACvB,OAAO,MAAM,OAAO,YAAY,wBAAwB,CAAC,GAAD,C;G;EAG5D,MAAM,mBAA
oB,GAAE,iB;IACxB,KAAK,KAAK,CAAC,MAAM,gBAAP,C;G;ECpFd,MAAM,eAAgB,GAAE,mB;IACpB,CA
AC,aAAc,GAAE,I;IACjB,OAAO,C;G;EAGX,MAAM,uBAAwB,GAAE,4C;IAC5B,MAAM,IAAK,GAAE,M;IAC
b,MAAM,IAAK,GAAE,M;IACb,MAAM,aAAc,GAAE,I;IACtB,OAAO,mBAAmB,CAAC,MAAD,EAAS,MAAT,
EAAiB,6BAA6B,CAAC,UAAD,CAA9C,C;G;EAG9B,iD;IACI,GAAG,WAAY,GAAE,sBAAsB,CAAC,OAAO,M
AAO,KAAI,UAAW,GAAE,KAAK,QAAP,GAAkB,KAAK,UAArD,C;IACvC,GAAG,YAAa,GAAE,G;IAClB,OA
AO,G;G;EAGX,IAAI,gCAAgC,CAChC,UACa,QAAS,IAAT,wBAAqC,Y;IAC1C,OAAO,MAAM,OAAO,QAAQ,k
B;GADvB,CADb,aAIe,QAAS,IAAT,wBAAqC,Y;IAC5C,OAAO,MAAM,OAAO,QAAQ,W;GADrB,CAJf,CADgC
,EAShC,UACa,QAAS,IAAT,wBAAqC,Y;IAC1C,OAAO,MAAM,OAAO,QAAQ,kB;GADvB,CADb,aAIe,QAAS,I
AAT,wBAAqC,Y;IAC5C,OAAO,MAAM,OAAO,QAAQ,W;GADrB,CAJf,CATgC,C;EAmBpC,uC;IACI,IAAI,KA
AK,MAAO,KAAI,IAApB,C;MACI,KAAK,MAAO,GAAE,aACE,CAAC,KAAK,qBAAqB,EAA3B,CADF,aAEC,I
AFD,aAGC,EAHD,cAIE,EAJF,SAKH,EALG,iBAMK,EANL,C;;IASlB,OAAO,KAAK,M;G;EChDhB,MAAM,QA
AS,GAAE,a;IACb,OAAoB,CAAZ,CAAE,GAAE,KAAQ,KAAG,EAAG,IAAG,E;G;EAGjC,MAAM,OAAQ,GAA
E,a;IACZ,OAAkB,CAAV,CAAE,GAAE,GAAM,KAAG,EAAG,IAAG,E;G;EAG/B,MAAM,OAAQ,GAAE,a;IAC
Z,OAAO,CAAE,GAAE,K;G;EAGf,MAAM,aAAc,GAAE,a;IAClB,OAAO,CAAE,YAAW,MAAM,KAAM,GAAE,
CAAF,GAAM,MAAM,KAAK,WAAW,CAAC,CAAD,C;G;EAGhE,MAAM,YAAa,GAAE,a;IACjB,OAAO,CAAE
,YAAW,MAAM,KAAM,GAAE,CAAC,MAAM,EAAT,GAAc,MAAM,YAAY,CAAC,CAAD,C;G;EAGpE,MAA
M,cAAe,GAAE,a;IACnB,OAAO,MAAM,QAAQ,CAAC,MAAM,YAAY,CAAC,CAAD,CAAnB,C;G;EAGzB,MA
AM,aAAc,GAAE,a;IAClB,OAAO,MAAM,OAAO,CAAC,MAAM,YAAY,CAAC,CAAD,CAAnB,C;G;EAGxB,M
AAM,eAAgB,GAAE,a;IACpB,OAAO,CAAC,C;G;EAGZ,MAAM,aAAc,GAAE,a;IAClB,OAAO,MAAM,OAAO,
CAAC,MAAM,YAAY,CAAC,CAAD,CAAnB,C;G;EAGxB,MAAM,YAAa,GAAE,a;IACjB,IAAI,CAAE,GAAE,U

AAR,C;MAAoB,OAAO,U;IAC3B,IAAI,CAAE,GAAE,WAAR,C;MAAqB,OAAO,W;IAC5B,OAAO,CAAE,GAAA
E,C;G;EAGf,MAAM,YAAa,GAAE,a;IACjB,IAAI,CAAE,IAAG,IAAT,C;MAAe,OAAO,C;IACtB,IAAI,CAAE,Y
AAW,MAAM,UAAvB,C;MAAmC,OAAO,C;IAC1C,OAAO,IAAI,MAAM,UAAV,CAAqB,CAArB,C;G;EAGX,M
AAM,UAAW,GAAE,a;IACf,IAAI,CAAE,IAAG,IAAT,C;MAAe,OAAO,C;IACtB,OAAO,MAAM,OAAO,CAAC,
CAAD,C;G;EClDxB,MAAM,OAAQ,GAAE,sB;IACZ,IAAI,IAAK,IAAG,IAAZ,C;MACI,OAAO,IAAK,IAAG,I;;I
AGnB,IAAI,IAAK,IAAG,IAAZ,C;MACI,OAAO,K;;IAGX,IAAI,IAAK,KAAI,IAAb,C;MACI,OAAO,IAAK,KAA
I,I;;IAGpB,IAAI,OAAO,IAAK,KAAI,QAAS,IAAG,OAAO,IAAI,OAAQ,KAAI,UAAvD,C;MACI,OAAO,IAAI,O
AAO,CAAC,IAAD,C;;IAGtB,IAAI,OAAO,IAAK,KAAI,QAAS,IAAG,OAAO,IAAK,KAAI,QAAhD,C;MACI,OA
AO,IAAK,KAAI,IAAK,KAAI,IAAK,KAAI,CAAE,IAAG,CAAE,GAAE,IAAK,KAAI,CAAE,GAAE,IAAnC,C;;I
AGzB,OAAO,IAAK,KAAI,I;G;EAGpB,MAAM,SAAU,GAAE,e;IACd,IAAI,GAAI,IAAG,IAAX,C;MACI,OAAO,
C;;IAEX,IAAI,UAAU,OAAO,G;IACrB,IAAI,QAAS,KAAI,OAAjB,C;MACI,OAAO,UAAW,KAAI,OAAO,GAA
G,SAAU,GAAE,GAAG,SAAS,EAAd,d,GAAmB,B,iBAAiB,CAAC,GAAD,C;;IAElF,IAAI,UAAW,KAAI,OAAnB,C;
MACI,OAAO,iBAAiB,CAAC,GAAD,C;;IAE5B,IAAI,QAAS,KAAI,OAAjB,C;MACI,OAAO,MAAM,eAAe,CAA
C,GAAD,C;;IAEhC,IAAI,SAAU,KAAI,OAAlB,C;MACI,OAAO,MAAM,CAAC,GAAD,C;;IAGjB,IAAI,MAAM,
MAAM,CAAC,GAAD,C;IAChB,OAAO,iBAAiB,CAAC,GAAD,C;G;EAI5B,MAAM,SAAU,GAAE,a;IACd,IAAI,
CAAE,IAAG,IAAT,C;MACI,OAAO,M;WAEN,IAAI,MAAM,WAAW,CAAC,CAAD,CAAR,C;MACD,OAAO,O
;;MAGP,OAAO,CAAC,SAAS,E;;G;EAKzB,IAAI,WAAW,a;EAGf,iCAAiC,sB;EAErC,gC;IACI,IAAI,EAAE
,8BAA+B,IAAG,GAAG,pC,CAAJ,C;MACI,IAAI,OAAQ,IAAI,OAAO,EAAG,GAAE,QAAU,GAAE,C;MACxC,MA
AM,eAAe,CAAC,GAAD,EAAM,8BAAN,EAAsC,QAAU,IAAV,cAA4B,KAA5B,CAAtC,C;;IAEzB,OAAO,GAA
G,CAAC,8BAAD,C;G;EAGd,gC;IACI,IAAI,OAAO,C;IACX,KAAK,IAAI,IAAI,CAAb,IAAI,CAAb,EAAgB,EAAG,GAAE,WAAW,CAAC,CAAD,C;MAC/B,IAAI,OAAQ,GAAE,WAAW,CAAC,CAAD,C;MACzB,IAAI,
OAAO,IAAK,KAAI,OAAO,IAAK,KAAI,OAAO,IAAK,KAAI,

AC,KAAD,CAAvB,C;;;IAGvB,OAAO,M;G;EAcT,MAAM,KAAK,gBAAiB,GAAE,CAAE,IAAG,E;EAOnC,MA
AM,KAAK,gBAAiB,GAAE,CAAE,IAAG,E;EAOnC,MAAM,KAAK,gBAAiB,GACxB,MAAM,KAAK,gBAAiB,
GAAE,MAAM,KAAK,gB;EAO7C,MAAM,KAAK,gBAAiB,GACxB,MAAM,KAAK,gBAAiB,GAAE,C;EAOlC,
MAAM,KAAK,gBAAiB,GACxB,MAAM,KAAK,gBAAiB,GAAE,MAAM,KAAK,gB;EAO7C,MAAM,KAAK,gB
AAiB,GACxB,MAAM,KAAK,gBAAiB,GAAE,MAAM,KAAK,gB;EAO7C,MAAM,KAAK,gBAAiB,GACxB,MA
AM,KAAK,gBAAiB,GAAE,C;EAIlC,MAAM,KAAK,KAAM,GAAE,MAAM,KAAK,QAAQ,CAAC,CAAD,C;EA
ItC,MAAM,KAAK,IAAK,GAAE,MAAM,KAAK,QAAQ,CAAC,CAAD,C;EAIrC,MAAM,KAAK,QAAS,GAAE,
MAAM,KAAK,QAAQ,CAAC,EAAD,C;EAIzC,MAAM,KAAK,UAAW,GAClB,MAAM,KAAK,SAAS,CAAC,aA
AW,GAAE,CAAd,EAAiB,UAAW,GAAE,CA9B,C;EAIxB,MAAM,KAAK,UAAW,GAAE,MAAM,KAAK,SA
S,CAAC,CAAD,EAAI,aAAW,GAAE,CAjB,C;EAO5C,MAAM,KAAK,YAAa,GAAE,MAAM,KAAK,QAAQ,C
AAC,CAAE,IAAG,EAAN,C;EAI7C,MAAM,KAAK,UAAU,MAAO,GAAE,Y;IAC5B,OAAO,IAAI,K;G;EAKb,M
AAM,KAAK,UAAU,SAAU,GAAE,Y;IAC/B,OAAO,IAAI,MAAO,GAAE,MAAM,KAAK,gBAAiB,GAAzC,IAAI
,mBAAmB,E;G;EAIhC,MAAM,KAAK,UAAU,SAAU,GAAE,Y;IAC/B,OAAO,IAAI,MAAO,GAAE,IAAI,K;G;E
AQ1B,MAAM,KAAK,UAAU,SAAU,GAAE,qB;IAC/B,IAAI,QAAQ,SAAU,IAAG,E;IACzB,IAAI,KAAM,GAAE
,CAAE,IAAG,EAAG,GAAE,KAAtB,C;;IAGb,IAAI,IAAI,OAAO,EAAf,C;MACE,OAAO,G;;IAGT,IAAI,IAAI,WAAW,EAAnB,C;MACE,IAAI,IAAI,WAAW,CAAC,
MAAM,KAAK,UAAZ,CAAnB,C;QAGE,IAAI,YAAY,MAAM,KAAK,WAAW,CAAC,KAAD,C;QACtC,IAAI,M
AAM,IAAI,IAAI,CAAC,SAAD,C;QAClB,IAAI,MAAM,GAAG,SAAS,CAAC,SAAD,CAAW,SAAS,CAAC,IAA
D,C;QAC1C,OAAO,GAAG,SAAS,CAAC,KAAD,CAAQ,GAAE,GAAG,MAAM,EAAE,SAAS,CAAC,KAAD,C;;
QAEjD,OAAO,GAAI,GAAE,IAAI,OAAO,EAAE,SAAS,CAAC,KAAD,C;;IAMvC,IAAI,eAAe,MAAM,KAAK,
WAAW,CAAC,IAAI,IAAI,CAAC,KAAD,C;IAEzC,IAAI,MAAM,I;IACV,IAAI,SAAS,E;IACb,OAAO,IAAP,C;MACE,IAAI,SAAS,GAAG,IAAI,CAAC,YAAD,C;MACpB,IAAI,SAAS,GAAG,SAAS,CAAC,
MAAM,SAAS,CAAC,YAAD,CAAhB,C;MAEb,E;MAC9D,IAAI,SAAS,MAAM,E;MACnB,IAAI,KAAK,GAAG,IAAI,
MAAM,KAAK,GAAG,IAAI,CAAC,KAAD,C;MAE9B,IAAI,KAAK,GAAG,IAAI,MAAM,KAAK,GAAG,IAAI,
CAAC,KAAD,C;;QAGlB,IAAI,KAAK,GAAG,IAAI,MAAM,KAAK,GAAG,IAAI,CAAC,KAAD,C;;;IAMtC,IAAI,
MAAM,IAAI,eAAe,IAAI,WAAW,CAAC,IAAI,IAAI,CAAC,KAAD,C;IAEjD,IAAI,MAAM,I;IACV,IAAI,KAAK,
GAAG,IAAI,CAAC,KAAD,C;IAEZ,IAAI,KAAK,GAAG,IAAI,MAAM,KAAK,GAAG,IAAI,CAAC,KAAD,C;;
IAGlB,IAAI,KAAK,GAAG,IAAI,MAAM,KAAK,GAAG,IAAI,CAAC,KAAD,C;;;IAMtC,IAAI,MAAM,IAAI,

E,IAAI,CAAC,MAAM,KAAK,IAAZ,C;;G;EAUzB,MAAM,KAAK,UAAU,IAAK,GAAE,iB;IAG1B,IAAI,MAAM
,IAAI,MAAO,KAAI,E;IACzB,IAAI,MAAM,IAAI,MAAO,GAAE,K;IACvB,IAAI,MAAM,IAAI,KAAM,KAAI,E;I
ACxB,IAAI,MAAM,IAAI,KAAM,GAAE,K;IAEtB,IAAI,MAAM,KAAK,MAAO,KAAI,E;IAC1B,IAAI,MAAM,K
AAK,MAAO,GAAE,K;IACxB,IAAI,MAAM,KAAK,KAAM,KAAI,E;IACzB,IAAI,MAAM,KAAK,KAAM,GAA
E,K;IAEvB,IAAI,MAAM,CAAV,,EAAa,MAAM,CAAnB,,EAAsB,MAAM,CA5B,EA+B,MAAM,C;IACrC,GA
AI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GA
AE,G;IACb,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,I
AAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,K;IACP,OAAO
,MAAM,KAAK,SAAS,CAAE,GAAI,IAAG,EAAI,GAAE,GAAf,,EAAqB,GAAI,IAAG,EAAI,GAAE,GAAlC,C;G;
EAS7B,MAAM,KAAK,UAAU,SAAU,GAAE,iB;IAC/B,OAAO,IAAI,IAAI,CAAC,KAAK,OAAO,EAAb,C;G;EA
SjB,MAAM,KAAK,UAAU,SAAU,GAAE,iB;IAC/B,IAAI,IAAI,OAAO,EAAf,C;MACE,OAAO,MAAM,KAAK,K
;WACb,IAAI,KAAK,OAAO,EAAhB,C;MACL,OAAO,MAAM,KAAK,K;;IAGpB,IAAI,IAAI,WAAW,CAAC,MA
AM,KAAK,UAAZ,,CAAnB,C;MACE,OAAO,KAAK,MAAM,EAAG,GAAE,MAAM,KAAK,UAAb,,GA0B,MA
AM,KAAK,K;WACrD,IAAI,KAAK,WAAW,CAAC,MAAM,KAAK,UAAZ,,CAApB,C;MACL,OAAO,IAAI,MA
AM,EAAG,GAAE,MAAM,KAAK,UAAb,,GA0B,MAAM,KAAK,K;;IAG3D,IAAI,IAAI,WAAW,EAAnB,C;MA
CE,IAAI,KAAK,WAAW,EAApB,C;QACE,OAAO,IAAI,OAAO,EAAE,SAAS,CAAC,KAAK,OAAO,EAAb,C;;Q
AE7B,OAAO,IAAI,OAAO,EAAE,SAAS,CAAC,KAAD,,CAAO,OAAO,E;;WAExC,IAAI,KAAK,WAAW,EAApB
,C;MACL,OAAO,IAAI,SAAS,CAAC,KAAK,OAAO,EAAb,,CAAgB,OAAO,E;;IAI7C,IAAI,IAAI,SAAS,CAAC,
MAAM,KAAK,YAAZ,,CA0B,IACvC,KAAK,SAAS,CAAC,MAAM,KAAK,YAAZ,,CADlB,C;MAEE,OAAO,M
AAM,KAAK,WAAW,CAAC,IAAI,SAAS,EAAG,GAAE,KAAK,SAAS,EAAjC,C;;IAM/B,IAAI,MAAM,IAAI,M
AAO,KAAI,E;IACzB,IAAI,MAAM,IAAI,MAAO,GAAE,K;IACvB,IAAI,MAAM,IAAI,KAAM,KAAI,E;IACxB,I
AAI,MAAM,IAAI,KAAM,GAAE,K;IAEtB,IAAI,MAAM,KAAK,MAAO,KAAI,E;IAC1B,IAAI,MAAM,KAAK,
MAAO,GAAE,K;IACxB,IAAI,MAAM,KAAK,KAAM,KAAI,E;IACzB,IAAI,MAAM,KAAK,KAAM,GAAE,K;I
AEvB,IAAI,MAAM,CAAV,,EAAa,MAAM,CAAnB,,EAAsB,MAAM,CA5B,EA+B,MAAM,C;IACrC,GAAI,IA
AG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;I
ACb,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,G
AAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAAI,E;IACf,G
AAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,
GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,
GAAE,GAAI,GAAE,GAAI,GAAE,GAAI,GAAE,GAAI,GAAE,GAAI,GAAE,GAAI,GAAE,G;IACjD,GAAI,IAA
G,K;IACP,OAAO,MAAM,KAAK,SAAS,CAAE,GAAI,IAAG,EAAI,GAAE,GAAf,,EAAqB,GAAI,IAAG,EAAI,G
AAE,GAAlC,C;G;EAS7B,MAAM,KAAK,UAAU,IAAK,GAAE,iB;IAC/B,IAAI,KAAK,OAAO,EAAhB,C;MACE
,MAAM,KAAK,CAAC,kBAAD,,C;WACN,IAAI,IAAI,OAAO,EAAf,C;MACL,OAAO,MAAM,KAAK,K;;IAGpB,
IAAI,IAAI,WAAW,CAAC,MAAM,KAAK,UAAZ,,CAAnB,C;MACE,IAAI,KAAK,WAAW,CAAC,MAAM,KAA
K,IAAZ,,CAAkB,IAAlC,C,KAAK,WAAW,CAAC,MAAM,KAAK,QAAZ,,CADpB,C;QAEE,OAAO,MAAM,KAAK,
U;aACb,IAAI,KAAK,WAAW,CAAC,MAAM,KAAK,UAAZ,,CAApB,C;QACL,OAAO,MAAM,KAAK,I;;QAGlB,
IAAI,WAAW,IAAI,WAAW,CAAC,CAAD,,C;QAC9B,IAAI,SAAS,QAAQ,IAAI,CAAC,KAAD,,CAAO,CAAP,
AAC,CAAD,,C;QAC1C,IAAI,MAAM,WAAW,CAAC,MAAM,KAAK,KAAZ,,CAAmB,CAArB,C;UACE,OAAO,KAAK,W
AAW,EAAG,GAAE,MAAM,KAAK,IAAb,,GAAoB,MAAM,KAAK,Q;;UAEzD,IAAI,MAAM,IAAI,SAAS,CAAC,
KAAK,SAAS,CAAC,MAAD,,CAAO,CAAf,,C;UACvB,IAAI,SAAS,MAAM,IAAI,CAAC,GAAG,IAAI,CAAC,KAD
AR,C;UACvB,OAAO,M;;;WAGN,IAAI,KAAK,WAAW,CAAC,MAAM,KAAK,UAAZ,,CAApB,C;MACL,OAAO
,MAAM,KAAK,K;;IAGpB,IAAI,IAAI,WAAW,EAAnB,C;MACE,IAAI,KAAK,WAAW,EAApB,C;QACE,OAAO,
IAAI,OAAO,EAAE,SAAS,CAAC,KAAK,OAAO,EAAb,C;;QAExB,OAAO,IAAI,OAAO,EAAE,SAAS,CAAC,KAA
D,,CAAO,OAAO,E;;WAEnC,IAAI,KAAK,WAAW,EAApB,C;MACL,OAAO,IAAI,IAAI,CAAC,KAAK,OAAO,E
AAb,,CAAgB,OAAO,E;;IAQxC,IAAI,MAAM,MAAM,KAAK,K;IACrB,IAAI,MAAM,I;IACV,IAAI,OAAO,GAG,mB
AAmB,CAAC,KAAD,,CA7B,C;MAGE,IAAI,SAAS,IAAI,IAAI,CAAC,CAAD,,EAAI,IAAI,MAAM,CAAC,CAA
G,SAAS,EAAG,GAAE,KAAK,SAAS,EAAjC,C;;IAMhC,IAAI,MAAM,IAAI,SAAS,EAAG,GAAE,MAAM,IAAI,
AAC,CAAD,,CAAS,GAAE,MAAM,IAAI,IAAI,CAAC,CAAD,,C;MAEvC,IAAI,QAAQ,IAAI,IAAI,CAAC,CAAD,,
AC,CAAD,,CAAS,GAAE,MAAM,IAAI,IAAI,CAAC,CAAD,,C;;IAIhC,IAAI,MAAM,IAAI,IAAI,CAAC,CAAD,,
```
```

AAI,CAAC,CAAD,EAAI,IAAK,GAAE,EAAX,C;MAIvC,IAAI,YAAY,MAAM,KAAK,WAAW,CAAC,MAAD,C;MACtC,IAAI,YAAY,SAAS,SAAS,CAAC,KAAD,C;MAClC,OAAO,SAAS,WAAW,EAAG,IAAG,SAAS,YAAY,CAAC,GAAD,CAAtD,C;QACE,MAAO,IAAAG,K;QACV,SAAU,GAAAE,E,MAAM,KAAK,WAAW,CAAC,MAAD,C;QAClC,SAAU,GAAAE,E,SAAS,SAAS,CAAC,KAAD,C;;MAKhC,IAAI,SAAS,OAAO,EAAApB,C;QACE,SAAU,GAAAE,E,MAAM,KAAK,I;;MAGzB,GAAAE,E,GAAG,IAAI,CAAC,SAAD,C;MACb,GAAAE,E,GAAG,SAAS,CAAC,SAAD,C;;IAEpB,GAAAE,E,GAAG,EAAS,MAAM,KAAK,UAAU,OAAO,GAAAE,E,iB;IAC7B,OAAO,IAAI,SAAS,CAAC,IAAI,IAAI,CAAC,IAAI,IAAI,CAAC,KAAD,C;MAC1C,GAAAE,E,GAAG,EAAK,CAAC,SAAS,CAAC,KAAD,C;QAAS,GAAAE,E,MAAM,KAAK,UAAU,IAAI,GAAAE,E,Y;IAC1C,OAAO,MAAM,KAAK,SAAS,CAAC,IAAI,IAAI,CAAC,KAAK,EAAG,IAAG,SAAS,CAAC,KAAD,C;MAC1E

C;;MAEjC,OAAO,MAAM,mBAAmB,CAAC,CAAD,EAAI,CAAJ,C;;IAEpC,IAAI,KAAM,KAAI,QAAS,IAAG,K
AAM,KAAI,SAApC,C;MACI,OAAO,MAAM,mBAAmB,CAAC,CAAD,EAAI,CAAJ,C;;IAEpC,OAAO,CAAC,g
BAAgB,CAAC,CAAD,C;G;EAG5B,MAAM,mBAAoB,GAAE,gB;IACxB,OAAO,CAAE,GAAE,CAAE,GAAE,E
AAF,GAAO,CAAE,GAAE,CAAE,GAAE,CAAF,GAAM,C;G;EAGpC,MAAM,gBAAiB,GAAE,gB;IACrB,IAAI,C
AAE,GAAE,CAAR,C;MAAW,OAAO,E;IAClB,IAAI,CAAE,GAAE,CAAR,C;MAAW,OAAO,C;IAElB,IAAI,CA
AE,KAAI,CAAV,C;MACI,IAAI,CAAE,KAAI,CAAV,C;QAAa,OAAO,C;MAEpB,IAAI,KAAK,CAAE,GAAE,C;
MACb,OAAO,EAAG,KAAI,CAAE,GAAE,CAAE,GAAE,CAAF,GAAM,GAAO,EAAG,GAAE,CAAE,GAAE,EAAF,GA
AO,C;;IAG7C,OAAO,CAAE,KAAI,CAAE,GAAG,CAAE,KAAI,CAAE,GAAE,CAAF,GAAM,GAAM,CAAjB,GAAsB,E;
G;EAGzC,MAAM,QAAS,GAAE,iB;IACb,OAAO,MAAM,OAAO,CAAC,KAAK,GAAC,CAAP,C;G;EAGxB,MA
AM,QAAS,GAAE,iB;IACb,OAAO,MAAM,OAAO,CAAC,KAAK,GAAC,CAAP,C;G;EAGxB,MAAM,KAAM,G
AAE,IAAI,KAAM,IAAG,I;EAE3B,MAAM,aAAc,GAAE,I;EAEtB,oB;IACI,OAAyB,CAAhB,CAAE,GAAE,YAA
Y,KAAG,CAAE,GAAE,KAAP,CAAa,GAAe,CAAZ,CAAE,GAAE,KAAQ,CAAE,GAAE,CAAE,GAAE,CAAP,C;
GAAE,C;G;EA6DtE,CA1DD,Y;IACG,IAAI,MAAM,IAAI,WAAJ,CAAgB,CAAhB,C;IACV,IAAI,aAAa,IAAI,YA
AJ,CAAiB,GAAjB,C;IACjB,IAAI,aAAa,IAAI,YAAJ,CAAiB,GAAjB,C;IACjB,IAAI,WAAW,IAAI,UAAJ,CAAe,
GAAf,C;IACf,IAAI,WAAW,C;IACf,IAAI,YAAY,C;IAEhB,UAAU,CAAC,CAAD,CAAI,GAAE,E;IAChB,IAAI,Q
AAQ,CAAC,QAAD,CAAW,KAAI,CAA3B,C;MACI,QAAS,GAAE,C;MACX,SAAU,GAAE,C;;IAGhB,MAAM,a
AAc,GAAE,iB;MAClB,OAAO,MAAM,gBAAgB,CAAC,KAAK,CAAC,KAAD,CAAQ,GAAE,GAAF,GAAM,KA
AtB,C;K;IAGjC,MAAM,gBAAiB,GAAE,iB;MACrB,UAAU,CAAC,CAAD,CAAI,GAAE,K;MAChB,OAAO,MA
AM,KAAK,SAAS,CAAC,QAAQ,CAAC,QAAD,CAAW,EAAqB,QAAQ,CAAC,SAAD,CAA7B,C;K;IAG/B,MAA
M,eAAgB,GAAE,iB;MACpB,QAAQ,CAAC,QAAD,CAAW,GAAE,KAAK,K;MAC1B,QAAQ,CAAC,SAAD,CA
AY,GAAE,KAAK,M;MAC3B,OAAO,UAAU,CAAC,CAAD,C;K;IAGrB,MAAM,YAAa,GAAE,iB;MACjB,OAA
O,MAAM,eAAe,CAAC,KAAK,CAAC,KAAD,CAAQ,GAAE,GAAF,GAAM,KAAtB,C;K;IAGhC,MAAM,eAAgB
,GAAE,iB;MACpB,UAAU,CAAC,CAAD,C;K;IAGd,MAAM,cAAe,GAAE,iB;MACnB,UAAU,CAAC,CAAD,CAAI,
GAAE,K;MAChB,QAAQ,CAAC,QAAD,CAAW,CAAX,C;K;IAGd,MAAM,cAAe,GAAE,iB;MACnB,QAAQ,CAAC,
CAAD,CAAI,GAAE,K;MACd,OAAO,UAAU,CAAC,CAAD,C;K;IAGrB,MAAM,cAAe,GAAE,iB;MACnB,UAAU,CAAC,
CAAD,CAAI,

,IAAI,CAAC,CAAC,CAAE,GAAE,IAAI,IAAV,C;UACnC,OAAgB,CAAR,CAAE,GAAE,EAAI,IAAE,C;;O;;IAI9
B,IAAI,OAAO,IAAI,KAAM,KAAI,WAAzB,C;MACI,IAAI,KAAM,GAAE,a;QACR,IAAI,IAAI,IAAI,IAAI,CAA
C,CAAD,C;QAChB,IAAI,KAAK,CAAE,GAAE,C;QACb,IAAI,CAAC,QAAQ,CAAC,CAAD,CAAI,IAAG,CAAC
,QAAQ,CAAC,EAAD,CAA7B,C;UAAmC,OAAO,IAAI,IAAI,CAAC,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,IA
AI,IAAnB,C;QAClD,OAAgB,CAAR,CAAE,GAAE,EAAI,IAAE,C;O;;IAI1B,IAAI,OAAO,IAAI,KAAM,KAAI,W
AAzB,C;MACI,IAAI,KAAM,GAAE,a;QACR,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAlB,C;UACI,IAAI
,SAAS,C;UACb,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAlB,C;YACI,MAAO,IAAI,CAAE,GAAE,CAA
E,GAAE,CAAG,GAAE,C;;UAE5B,OAAO,M;;UAGP,IAAI,IAAI,IAAI,IAAI,CAAC,CAAC,CAAF,CAAhB,B,EAAs
B,IAAI,IAAI,IAAI,CAAC,CAAC,CAAF,C;UAClC,OAAO,CAAE,KAAI,QAAS,GAAE,CAAF,GAAM,GAAE,CAAE,KA
AI,QAAS,GAAE,EAAF,GAAe,CAAP,CAAE,GAAE,CAAG,GAAE,CAAE,GAAE,CAAP,C;;O;;IAQtE,IAAI,OA
AO,IAAI,MAAO,KAAI,WA1B,C;MACI,IAAI,QAAQ,a;QACR,IAAI,CAAE,IAAG,CAAC,cAAV,C;UAEI,IAAI
,CAAE,GAAE,oBAAR,C;YAEI,IAAI,CAAE,GAAE,oBAAR,C;cAGI,OAAO,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,GA
AE,IAAI,I;;cAKzB,OAAO,IAAI,IAAI,CAAC,CAAE,GAAE,CAAE,GAAE,CAAE,GAAG,CAAE,IAAG,CAAE,GAAE,CAAP,C
AAZ,C;;;YAKnB,OAAO,IAAI,IAAI,CAAC,CAAE,GAAE,IAAI,KAAK,CAAC,CAAE,GAAE,CAAE,GAAE,CAAE,GAAE,CA
AT,CAAd,C;;eAGlB,IAAI,CAAE,IAAG,CAAC,cAAV,C;UAED,OAAO,CAAC,KAAK,CAAC,CAAC,CAAF,C;;
UAKb,IAAI,SAAS,C;UACb,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,IAAG,cAAnB,C;YAEI,IAAI,KAAK,CAAE,G
AAE,CAAE,GAAE,C;YAEjB,MAAO,IAAG,EAAG,GAAE,C;;UAEnB,OAAO,M;;O;MAGf,IAAI,MAAO,GAAE,
K;;IAEjB,IAAI,OAAO,IAAI,MAAO,KAAI,WA1B,C;MACI,IAAI,MAAO,GAAE,a;QACT,IAAI,CAAE,GAAE,
CAAR,C;UAEI,OAAO,G;eAEN,IAAI,CAAE,GAAE,CAAE,IAAG,cAAb,C;UAED,IAAI,CAAE,GAAE,oBAAR,C
;YAGI,OAAO,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,IAAI,I;;YAIzB,OAAO,IAAI,IAAI,CAAC,CAAE,GAAE,GAAE,I
AAI,KAAK,CAAC,CAAE,GAAE,GAAE,CAAE,GAAE,CAAT,CAAd,C;;;UAKnB,IAAI,IAAI,IAAI,KAAK,CAAC,CAAE,GAAE,CAA
E,GAAE,CAAL,C;UAEjB,IAAI,SAAS,C;UACb,IAAI,CAAE,IAAG,cAAb,C;YAEI,IAAI,KAAK,CAAE,GAAE,GAAE,C
AAE,GAAE,C;YAEjB,MAAO,IAAG,EAAG,GAAE,E;;UAGnB,OAAO,IAAI,KAAK,CAAC,CAAD,CAAI,GAAE,GAAE
,M;;O;;IAIlC,IAAI,OAAO,IAAI,MAAO,KAAI,WA1B,C;MACI,IAAI,MAAO,GAAE,a;QACT,IAAI,IAAI,IAAI,
CAAC,CAAD,CAAI,GAAE,cAAb,C;UAED,IAAI,SAAS,C;UACb,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,
cAAlB,C;YAEI,MAAO,IAAI,CAAE,GAAE,CAAE,GAAE,CAAE,GAAE,CAAG,GAAE,C;;UAEpC,OAAO,M;;QAEX,OAAO,I
AAI,IAAI,CAAS,CAAP,CAAE,GAAE,CAAE,GAAE,CAAG,GAAE,CAAE,GAAE,CAAP,CAAT,CAAoB,GAAE,C;O;;IAG7C,I
AAI,OAAO,IAAI,MAAO,KAAI,WA1B,C;MACI,IAAI,MAAO,GAAE,a;QACT,IAAI,IAAI,IAAI,CAAC,CAAD,
CAAI,GAAE,cAAb,C;UAED,IAAI,KAAK,CAAE,GAAE,GAAE,C;UACb,IAAI,CAAE,IAAG,KAAK,CAAE,GAAE,GAAE,C;UACd,IAAI,K
AAK,CAAE,GAAE,GAAE,C;UAEd,OAAO,CAAC,CAAE,GAAE,GAAE,CAAE,GAAE,CAAE,GAAE,GAAE,CAAE,GAAE,CAAE,GAAE,GAAE,G
AAE,CAAE,GAAE,C;;QAEvC,OAAO,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,GAAE,CAAL,C;O;;IAGvB,IAAI,OAAO,IAAI,MAAO,KAAI,W
AAzB,C;MACI,IAAI,MAAO,GAAE,a;QACT,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAb,C;UAED,IAAI,KAAK,CAAE,GAAE,GAAE,cAAR

AK,IAAG,IAAZ,C;QACI,MAAM,IAAI,SAAJ,CAAc,6BAAd,C;;MAGV,IAAI,IAAI,MAAM,CAAC,IAAD,C;MAGd,IAAI,MAAM,CAAC,OAAQ,KAAI,C;MAGvB,IAAI,QAAQ,SAAS,CAAC,CAAD,C;MACrB,IAAI,gBAAgB,KAAM,IAAG,C;MAG7B,IAAI,IAAI,aAAc,GAAE,CAAE,GAClB,IAAI,IAAI,CAAC,GAAI,GAAE,aAAP,EAAsB,CAAtB,CADU,GAElB,IAAI,IAAI,CAAC,aAAD,EAAgB,GAAhB,C;MAGhB,IAAI,MAAM,SAAS,CAAC,CAAD,C;MACnB,IAAI,cAAc,GAAI,KAAI,SAAU,GAClB,IAAI,GADc,GAAI,GAAE,IAAG,C;MAG/B,IAAI,aAAa,WAAY,GAAE,CAAE,GAChB,IAAI,IAAI,CAAC,GAAI,GAAE,WAAP,EAAoB,CAAC,AAAoB,CAApB,CADT,GAAI,GAAE,IAAG,C;MAGzC,IAAI,OAAO,CAAE,GAAE,GAAI,GAAE,IAAI,IAAI,CAAC,GAAI,GAAE,IAAG,C;MAGvC

Ac,YAAhB,GAA+B,I;IAC7E,OAAO,gBAAiB,IAAG,IAAK,IAAG,0BAA0B,CAAC,gBAAD,EAAmB,KAAnB,C;G;EASjE,MAAM,OAAQ,GAAE,yB;IACZ,IAAI,KAAM,KAAI,MAAd,C;MACI,QAAQ,OAAO,MAAf,C;aACS,Q;aACA,Q;aACA,S;aACA,U;UACD,OAAO,I;;UAEP,OAAO,MAAO,YAAW,M;;;IAIrC,IAAI,MAAO,IAAG,IAAK,IAAG,KAAM,KAAI,OAAO,MAAO,KAAI,QAAS,IAAG,OAAO,MAAO,KAAI,UApD,CAApC,C;MACI,OAAO,K;;IAGX,IAAI,OAAO,KAAM,KAAI,UAAW,IAAG,MAAO,YAAW,KArD,C;MACI,OAAO,I;;IAGX,IAAI,QAAQ,MAAM,eAAe,CAAC,KAAD,C;IACjC,IAAI,cAAc,IAAG,KAAM,IAAG,KAAM,GAAE,KAAK,YAAY,GAAE,I;IACzD,IAAI,WAAW,IAAG,KAAM,IAAG,KAAM,IAAG,YAAa,IAAG,MAAO,YAAW,KAAK,C;MACI,OAAO,I;;IAGX,IAAI,QAAQ,KAAM,KAAI,OAAO,KAAM,KAAI,UAAW

AAT,C;MAEvB,kBACuB,C;MAEvB,iBACsB,E;K;;;IAxB1B,4C;MAAA,2C;QAAA,0B;;MAAA,oC;K;IA2BA,8B;MAAA,kC;MAEI,iBACqB,W;MAErB,iBACqB,U;MAErB,kBACuB,C;MAEvB,iBACsB,E;K;;;IAZ1B,0C;MAAA,yC;QAAA,wB;;MAAA,kC;K;IAeA,+B;MAAA,mC;MAEI,iBACJ,MAAM,KAAoB,U;MAEtB,iBACJ,MAAM,KAAoB,U;MAEtB,kBACuB,C;MAEvB,iBACsB,E;K;;;IAZ1B,2C;MAAA,0C;QAAA,yB;;MAAA,mC;K;IAeA,gC;MAAA,oC;MAEI,iBACuB,U;MAEvB,iBACuB,K;MAEvB,kBACuB,C;MAEvB,iBACsB,E;K;;;IAZ1B,4C;MAAA,2C;QAAA,0B;;MAAA,oC;K;IAeA,+B;MAAA,mC;MAEI,iBACsB,Q;MAEtB,iBACsB,G;MAEtB,kBACuB,C;MAEvB,iBACsB,C;K;;;IAZ1B,2C;MAAA,0C;QAAA,yB;;MAAA,mC;K;IAeA,+B;MAAA,mC;MAEI,iBACmC,C;MAEnC,iBACmC,K;MAEnC,0BAC4C,K;MAE5C,0BAC4C,K;MAE5C,yBAC2C,K;MAE3C,yBAC2C,K;MAE3C,qBACuC,uB;MAEvC,qBACuC,sB;MAEvC,kBACuB,C;MAEvB,iBACsB,E;K;;;IA9B1B,2C;MAAA,0C;QAAA,yB;;MAAA,mC;K;IAiCA,iC;MAAA,qC;K;;;IAAA,6C;MAAA,4C;QAAA,2B;;MAAA,qC;K;IAEA,kC;MAAA,sC;K;;;IAAA,8C;MAAA,6C;QAAA,4B;;MAAA,sC;K;;;

C,UAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KAPjE,C;sGAUA,yB;MAAA,8D;MAAA,iD;QAOI,OAAW,SA
AS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KAPjE,C;sGAUA,yB;
MAAA,8D;MAAA,gC;MAAA,iD;QAOI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KAPjE,C;sGAUA,yB;MAAA,sD;MAAA,mC;QAOI,OAAY,UAAL,SAAK,EAAU,
KAAV,C;O;KAPhB,C;qGAUA,yB;MAAA,qD;MAAA,mC;QAOI,OAAY,UAAL,SAAK,EAAU,KAAV,C;O;KAP
hB,C;sGAUA,yB;MAAA,sD;MAAA,mC;QAOI,OAAY,UAAL,SAAK,EAAU,KAAV,C;O;KAPhB,C;sGAUA,yB;
MAAA,sD;MAAA,mC;QAOI,OAAY,UAAL,SAAK,EAAU,KAAV,C;O;KAPhB,C;sGAUA,yB;MAAA,sD;MAA
A,mC;QAOI,OAAY,UAAL,SAAK,EAAU,KAAV,C;O;KAPhB,C;sGAUA,yB;MAAA,sD;MAAA,mC;QAOI,OAA
Y,UAAL,SAAK,EAAU,KAAV,C;O;KAPhB,C;sGAUA,yB;MAAA,sD;MAAA,mC;QAOI,OAAY,UAAL,SAAK,E
AAU,KAAV,C;O;KAPhB,C;sGAUA,yB;MAAA,sD;MAAA,mC;QAOI,OAAY,UAAL,SAAK,EAAU,KAAV,C;O;
KAPhB,C;sGAUA,yB;MAAA,sD;MAAA,mC;QAOI,OAAY,UAAL,SAAK,EAAU,KAAV,C;O;KAPhB,C;8EAUA
,gC;MAOW,sB;;QAybS,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAzB,SAAyB,O,CAAU,OAAV,,CAAJ,C;YAAwB
,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;MA1bP,yB;K;gFAGJ,gC;MAOW,sB;;QAubS,Q;QAAhB,iD;UAAgB,
cAAhB,e;UAAsB,IAAvB,SAAuB,O,CAAU,OAAV,,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;M
AxbP,yB;K;gFAGJ,gC;MAOW,sB;;QAqbS,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAArB,SAAqB,O,CAAU,OAA
V,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;MAtbP,yB;K;gFAGJ,gC;MAOW,sB;;QAmbS,Q;
QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAnB,SAAmB,O,CAAU,OAAV,,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;Q
AC9C,qBAAO,I;;;MApbP,yB;K;gFAGJ,gC;MAOW,sB;;QAibS,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAjB,S
AibO,CAAU,OAAV,,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;MAlbP,yB;K;gFAGJ,gC;MAO
W,sB;;QA+aS,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA/aH,SA+aO,O,CAAU,OAAV,,CAAJ,C;YAAwB,qBAAO,
O;YAAP,uB;;;QAC9C,qBAAO,I;;;MAhbP,yB;K;gFAGJ,gC;MAOW,sB;;QA6aS,Q;QAAhB,iD;UAAgB,cAAhB,e;
UAAsB,IA7aH,SA6aO,O,CAAU,OAAV,,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;MA9aP,yB;K
;gFAGJ,gC;MAOW,sB;;QA2aS,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA3aH,SA2aO,O,CAAU,OAAV,,CAAJ,C;
YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;MA5aP,yB;K;gFAGJ,yB;MA4aO,oC;MAAA,gC;MA5aP,uC;
QAOW,sB;;UAyaS,Q;UAAhB,iD;YAAgB,cAAhB,0B;YAAsB,IAzaH,SAyaO,CAAU,oBAAV,,CAAJ,C;cAAwB,q
BAAO,O;cAAP,uB;;;UAC9C,qBAAO,I;;;QA1aP,yB;O;KAPJ,C;sFAUA,yB;MAAA,+C;MAAA,gC;MAAA,uC;QAOW,
sB;;UAuaS,Q;UAAhB,iD;YAAgB,cAAhB,0B;YAAsB,IAAtB,SAAsB,O,CAAU,oBAAV,,CAAJ,C;cAAwB,qBAAO,
O;cAAP,uB;;;UAC9C,qBAAO,I;;;QAxaP,yB;O;KAPJ,C;sFAUA,yB;MAAA,+C;MAAA,gC;MAAA,uC;QAOW,
sB;;UAqaS,Q;UAAhB,iD;YAAgB,cAAhB,0B;YAAsB,IAAtB,SAAsB,O,CAAU,oBAAV,,CAAJ,C;cAAwB,qBAAO,
O;cAAP,uB;;;UAC9C,qBAAO,I;;;QAtaP,yB;O;KAPJ,C;sFAUA,yB;MAAA,+C;MAAA,gC;MAAA,uC;QAOW,
sB;;UAmaS,Q;UAAhB,iD;YAAgB,cAAhB,0B;YAAsB,IAAtB,SAAsB,O,CAAU,oBAAV,,CAAJ,C;cAAwB,qBAAO,
O;cAAP,uB;;;UAC9C,qBAAO,I;;;QApaP,yB;O;KAPJ,C;sFAUA,yB;MAAA,+C;MAAA,gC;MAAA,uC;QAOW,
sB;;UAiaS,Q;UAAhB,iD;YAAgB,cAAhB,0B;YAAsB,IAAtB,SAAsB,O,CAAU,oBAAV,,CAAJ,C;cAAwB,qBAAO,
O;cAAP,uB;;;UAC9C,qBAAO,I;;;QAlaP,yB;O;KAPJ,C;sFAUA,yB;MAAA,+C;MAAA,gC;MAAA,uC;QAOW,
sB;;UA+aS,Q;UAAhB,iD;YAAgB,cAAhB,0B;YAAsB,IAAtB,SAAsB,O,CAAU,oBAAV,,CAAJ,C;cAAwB,qBAAO,
O;cAAP,uB;;;UAC9C,qBAAO,I;;;QAhbP,yB;O;KAPJ,C;sFAUA,yB;MAAA,+C;MAAA,gC;MAAA,uC;QAOW,
sB;;UA6aS,Q;UAAhB,iD;YAAgB,cAAhB,0B;YAAsB,IAAtB,SAAsB,O,CAAU,oBAAV,,CAAJ,C;cAAwB,qBAAO,
O;cAAP,uB;;;UAC9C,qBAAO,I;;;QA9aP,yB;O;KAPJ,C;sFAUA,yB;MAAA,+C;MAAA,gC;MAAA,uC;QAOW,
sB;;UA2aS,Q;UAAhB,iD;YAAgB,cAAhB,0B;YAAsB,IAAtB,SAAsB,O,CAAU,oBAAV,,CAAJ,C;cAAwB,qBAAO,
O;cAAP,uB;;;UAC9C,qBAAO,I;;;QA5aP,yB;O;KAPJ,C;sFAUA,yB;MAAA,+C;MAAA,gC;MAAA,uC;QAOW,
sB;;UAyaS,Q;UAAhB,iD;YAAgB,cAAhB,0B;YAAsB,IAAtB,SAAsB,O,CAAU,oBAAV,,CAAJ,C;cAAwB,qBAAO,

A,OAAa,SAAR,sBAAQ,CAAb,W;UAAd,OAAc,cAAd,C;YAAc,uB;YACV,cAAc,UAAK,KAAL,C;YACd,IAj3Cc,SAi3CV,CAAU,oBAAV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;;QAn3CP,wB;O;KAPJ,C;IAUA,0B;MAKI,IA4uNO,qBAAQ,CA5uNf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;K;IAGX,4B;MAKI,IA0uNO,qBAAQ,CA1uNf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;K;IAGX,4B;MAKI,IAwuNO,qBAAQ,CAxuNf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;K;IAGX,4B;MAKI,IAsuNO,qBAAQ,CAtuNf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;K;IAGX,4B;MAKI,IAouNO,qBAAQ,CApuNf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;K;IAGX,4B;MAKI,IAkuNO,qBAAQ,CAluNf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;K;IAGX,4B;MAKI,IAguNO,qBAAQ,CAhuNf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;K;IAGX,4B;MAKI,IA8tNO,qBAAQ,CA9tNf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;K;IAGX,4B;MAKI,IA4tNO,qBAAQ,CA5tNf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;K;kFAGX,yB;MAAA,iE;MAAA,uC;QAAA,oB,Q;QAAA,hB,wBAAA,gB,SAAA,hB,gB;UAAA,gB,cAAA,SAAA,hB,M;UAAA,sB,IAAA,I,UAAA,U,OAAA,V,CAAA,J,C;YAAA,wB,OAAA,O;;QAAA,rD,MAAA,gCAAA,uB,mDAAA,vB,C;O;KAAA,W,C;kFAAA,yB;MAAA,iE;MAAA,uC;QAAA,oB,Q;QAAA,hB,wBAAA,gB,SAAA,hB,gB;UAAA,gB,cAAA,SAAA,hB,M;UAAA,sB,IAAA,I,UAAA,U,OAAA,V,CAAA,J,C;YAAA,wB,OAAA,O;;QAAA,rD,MAAA,gCAAA,uB,mDAAA,vB,C;O;KAAA,W,C;m

O,I;K;+FAGX,gC;MAIoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OA
AV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;K;+FAGX,gC;MAIoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAA
gB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;K;+FAGX,yB;
MAAA,oC;MAAA,gC;MAAA,uC;QAIoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;U
AAsB,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,OAAO,O;;QACrD,OAAO,I;O;KALX,C;wFAQA,yB;MAAA,8D;M
AAA,iD;QAKI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,
C;O;KALjE,C;0FAQA,yB;MAAA,8D;MAAA,iD;QAKI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,UA
AI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KALjE,C;0FAQA,yB;MAAA,8D;MAAA,iD;QAKI,OAAW,SAAS,C
AAT,IAAc,SAAS,wBAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KALjE,C;0FAQA,yB;MAA
A,8D;MAAA,iD;QAKI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa
,KAAb,C;O;KALjE,C;0FAQA,yB;MAAA,8D;MAAA,iD;QAKI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GA
AsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KALjE,C;0FAQA,yB;MAAA,8D;MAAA,iD;QAKI,OAAW,
SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KALjE,C;0FAQA,y
B;MAAA,8D;MAAA,iD;QAKI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAs
D,aAAa,KAAb,C;O;KALjE,C;0FAQA,yB;MAAA,8D;MAAA,iD;QAKI,OAAW,SAAS,CAAT,IAAc,SAAS,wBA
A3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KALjE,C;0FAQA,yB;MAAA,8D;MAAA,gC;MAA
A,iD;QAKI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,uBAAa,KAAb,E;
O;KALjE,C;IAQA,qC;MAMI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa
,I;K;IAGjE,uC;MAMI,OAAW,SAAS,CAAT,IAAc,SAAS,0BAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,I;K;IA
GjE,uC;MAMI,OAAW,SAAS,CAAT,IAAc,SAAS,0BAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,I;K;IAGjE,uC;
MAMI,OAAW,SAAS,CAAT,IAAc,SAAS,0BAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,I;K;IAGjE,uC;MAMI,
OAAW,SAAS,CAAT,IAAc,SAAS,0BAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,I;K;IAGjE,uC;MAMI,OAAW,
SAAS,CAAT,IAAc,SAAS,0BAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,I;K;IAGjE,uC;MAMI,OAAW,SAAS,
CAAT,IAAc,SAAS,0BAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,I;K;IAGjE,uC;MAMI,OAAW,SAAS,CAAT,I
AAc,SAAS,0BAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,I;K;IAGjE,uC;MAMI,OAAW,SAAS,CAAT,IAAc,SA
AS,0BAA3B,GAAsC,UAAI,KAAJ,CAAtC,GAAsD,aAAa,I;K;IAGjE,qC;MAOI,IAAI,eAAJ,C;QAAI,wD;UACI,IAAI,UA
AK,KAAL,SAAJ,C;YACI,OAAO,K;;;;QAIf,8D;UACI,IAAI,gBAAW,UAAK,OAAL,CAAX,CAAJ,C;YACI,OAA
O,O;;;;MAInB,OAAO,E;K;IAGX,uC;MAOI,wD;QACI,IAAI,YAAW,UAAK,KAAL,CAAf,C;UACI,OAAO,K;;;M
AGf,OAAO,E;K;IAGX,uC;MAOI,wD;QACI,IAAI,YAAW,UAAK,KAAL,CAAf,C;UACI,OAAO,K;;;MAGf,OAA
O,E;K;IAGX,uC;MAOI,wD;QACI,IAAI,YAAW,UAAK,KAAL,CAAf,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;IAG
X,uC;MAOI,wD;QACI,IAAI,gBAAW,UAAK,KAAL,CAAX,CAAJ,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;IAGX,
uC;MAOI,wD;QACI,IAAI,YAAW,UAAK,KAAL,CAAf,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;IAGX,uC;MAOI
,wD;QACI,IAAI,YAAW,UAAK,KAAL,CAAf,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;IAGX,uC;MAOI,wD;QACI
,IAAI,YAAW,UAAK,KAAL,CAAf,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;IAGX,uC;MAOI,wD;QACI,IAAI,YAA
W,UAAK,KAAL,CAAf,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;8FAGX,gC;MAOI,wD;QACI,IAAI,UAAU,UAAK
,KAAL,CAAV,CAAJ,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;gGAGX,gC;MAOI,wD;QACI,IAAI,UAAU,UAAK,K
AAL,CAAV,CAAJ,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;gGAGX,gC;MAOI,wD;QACI,IAAI,UAAU,UAAK,KAAL,
CAAV,CAAJ,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;gGAGX,gC;MAOI,wD;QACI,IAAI,UAAU,UAAK,KAAL,CAAV
,CAAJ,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;gGAGX,gC;MAOI,wD;QACI,IAAI,UAAU,UAAK,KAAL,CAAV,C
AAJ,C;UACI,OAAO,K;;;MAGf,OAAO,E;K;gGAGX,gC;MAOI,wD;QACI,IAAI,UAAU,UAAK,KAAL,CAAV,CAAJ,
C;UACI,OAAO,K;;;MAGf,OAAO,E;K;gGAGX,gC;MAOI,wD;QACI,IAAI,UAAU,UAAK,KAAL,CAAV,CAAJ,C;UA
CI,OAAO,K;;;MAGf,OAAO,E;K;gGAGX,yB;MAAA,oC;MAAA,uC;QAII,wD;UACI,IAAI,UAAU,sBAAK,KAAL,,EAAV,
CAAJ,C;YACI,OAAO,K;;;QAIf,gD;UAAI,OAAO,E;O;KATX,C;4FAYA,yB;MAAA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,
OAAQ,SAAR,sBAAQ,CAAR,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,UAAK,KAAL,,CAAV,CAAJ,C;YACI,OAAO,K;;;QAIf,gD;UAAI,OAAO,E;O;KATX,C;8FAYA,yB;MAAA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,IA
AI,UAAU,UAAK,KAAL,,CAAV,CAAJ,C;YACI,OAAO,K;;;QAIf,gD;UAAI,OAAO,E;O;KATX,C;8FAYA,yB;MAAA,0D;

MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAQ,SAAR,sBAAQ,CAAR,W;QAAd,OAAc,cAAd,C;UAAc,uB;UAC
V,IAAI,UAAU,UAAK,KAAL,CAAV,CAAJ,C;YACI,OAAO,K;;;QAGf,OAAO,E;O;KATX,C;8FAYA,yB;MAAA,
0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAQ,SAAR,sBAAQ,CAAR,W;QAAd,OAAc,cAAd,C;UAAc,uB;U
ACV,IAAI,UAAU,UAAK,KAAL,CAAV,CAAJ,C;YACI,OAAO,K;;;QAGf,OAAO,E;O;KATX,C;8FAYA,yB;MA
AA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAQ,SAAR,sBAAQ,CAAR,W;QAAd,OAAc,cAAd,C;UAAc,u
B;UACV,IAAI,UAAU,UAAK,KAAL,CAAV,CAAJ,C;YACI,OAAO,K;;;QAGf,OAAO,E;O;KATX,C;8FAYA,yB;
MAAA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAQ,SAAR,sBAAQ,CAAR,W;QAAd,OAAc,cAAd,C;UA
Ac,uB;UACV,IAAI,UAAU,UAAK,KAAL,CAAV,CAAJ,C;YACI,OAAO,K;;;QAGf,OAAO,E;O;KATX,C;8FAYA
,yB;MAAA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAQ,SAAR,sBAAQ,CAAR,W;QAAd,OAAc,cAAd,C;
UAAc,uB;UACV,IAAI,UAAU,UAAK,KAAL,CAAV,CAAJ,C;YACI,OAAO,K;;;QAGf,OAAO,E;O;KATX,C;8FA
YA,yB;MAAA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAQ,SAAR,sBAAQ,CAAR,W;QAAd,OAAc,cAAd
,C;UAAc,uB;UACV,IAAI,UAAU,UAAK,KAAL,CAAV,CAAJ,C;YACI,OAAO,K;;;QAGf,OAAO,E;O;KATX,C;8
FAYA,yB;MAAA,0D;MAAA,+C;MAAA,oC;MAAA,uC;QAIkB,Q;QAAA,OAAQ,SAAR,sBAAQ,CAAR,W;QA
Ad,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,sBAAK,KAAL,,EAAV,CAAJ,C;YACI,OAAO,K;;;QAGf,OAA
O,E;O;KATX,C;IAYA,yB;MAQI,IAg7LO,qBAAQ,CAh7Lf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO
,UAAK,wBAAL,C;K;IAGX,2B;MAQI,IA26LO,qBAAQ,CA36Lf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,
OAAO,UAAK,0BAAL,C;K;IAGX,2B;MAQI,IAs6LO,qBAAQ,CAt6Lf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MA
CV,OAAO,UAAK,0BAAL,C;K;IAGX,2B;MAQI,IAi6LO,qBAAQ,CAj6Lf,C;QACI,MAAM,2BAAuB,iBAAvB,C;
MACV,OAAO,UAAK,0BAAL,C;K;IAGX,2B;MAQI,IA45LO,qBAAQ,CA55Lf,C;QACI,MAAM,2BAAuB,iBAA
vB,C;MACV,OAAO,UAAK,0BAAL,C;K;IAGX,2B;MAQI,IAu5LO,qBAAQ,CAv5Lf,C;QACI,MAAM,2BAAuB,iB
BAAvB,C;MACV,OAAO,UAAK,0BAAL,C;K;IAGX,2B;MAQI,IAk5LO,qBAAQ,CAl5Lf,C;QACI,MAAM,2BA
AuB,iBAAvB,C;MACV,OAAO,UAAK,0BAAL,C;K;IAGX,2B;MAQI,IA64LO,qBAAQ,CA74Lf,C;QACI,MAAM
,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,0BAAL,C;K;IAGX,2B;MAQI,IAw4LO,qBAAQ,CAx4Lf,C;QACI,M
AAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,0BAAL,C;K;gFAGX,yB;MAAA,0D;MAAA,+C;MAAA,iE;M
AAA,uC;QAQkB,Q;QAAA,OAAa,a,SAAR,YAAL,,SAAK,CAAQ,CAAb,b,W;QAAd,OAAc,cAAd,C;UAAc,uB;UAC
V,cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEnC,MAAM,gCAAuB,mD
AAvB,C;O;KAZV,C;gFAeA,yB;MAAA,0D;MAAA,+C;MAAA,iE;MAAA,uC;QAQkB,Q;QAAA,OAAa,a,SAAR,Y
AAL,SAAK,CAAQ,CAAb,b,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,UAA
U,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEnC,MAAM,gCAAuB,mDAAvB,C;O;KAZV,C;iFAeA,yB;MAAA,0D;
MAAA,+C;MAAA,iE;MAAA,uC;QAQkB,Q;QAAA,OAAa,a,SAAR,YAAL,SAAK,CAAQ,CAAb,b,W;QAAd,OAAc,
cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEn
C,MAAM,gCAAuB,mDAAvB,C;O;KAZV,C;iFAeA,yB;MAAA,0D;MAAA,+C;MAAA,iE;MAAA,uC;QAQkB,Q;
QAAA,OAAa,a,SAAR,YAAL,SAAK,CAAQ,CAAb,b,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAA
L,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEnC,MAAM,gCAAuB,mDAAvB,C;O;KAZV,C;i
FAeA,yB;MAAA,0D;MAAA,+C;MAAA,iE;MAAA,uC;QAQkB,Q;QAAA,OAAa,a,SAAR,YAAL,SAAK,CAAQ,C
AAb,b,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YA
AwB,OAAO,O;;QAEnC,MAAM,gCAAuB,mDAAvB,C;O;KAZV,C;iFAeA,yB;MAAA,0D;MAAA,+C;MAAA,iE;
MAAA,uC;QAQkB,Q;QAAA,OAAa,a,SAAR,YAAL,SAAK,CAAQ,CAAb,b,W;QAAd,OAAc,cAAd,C;UAAc,uB;UA
CV,cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEnC,MAAM,gCAAuB,m
DAAvB,C;O;KAZV,C;iFAeA,yB;MAAA,0D;MAAA,+C;MAAA,iE;MAAA,uC;QAQkB,Q;QAAA,OAAa,a,SAAR,
YAAL,SAAK,CAAQ,CAAb,b,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,UA
AU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEnC,MAAM,gCAAuB,mDAAvB,C;O;KAZV,C;iFAeA,yB;MAAA,0
D;MAAA,+C;MAAA,iE;MAAA,uC;QAQkB,Q;QAAA,OAAa,a,SAAR,YAAL,SAAK,CAAQ,CAAb,b,W;QAAd,OAA
c,cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QA
EnC,MAAM,gCAAuB,mDAAvB,C;O;KAZV,C;iFAeA,yB;MAAA,0D;MAAA,+C;MAAA,oC;MAAA,iE;MAAA,
uC;QAQkB,Q;QAAA,OAAa,a,SAAR,YAAL,SAAK,CAAQ,CAAb,b,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,cAA
c,UAAK,KAAL,C;UACd,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,OAAO,O;;QAEnC,MAAM,gCAAuB,mDAAvB
,C;O;KAZV,C;IAeA,yC;MAKI,sBAAK,KAAL,,EAAK,CAAQ,CAAb,b,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,
cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEnC,MAAM,gCAAuB,mDAAvB,C;

Ad,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAK,KAAL,SAAJ,C;YACI,OAAO,K;;;;QAID,SAAQ,WAAR,sBAAAQ,CAAR,W;QAAd,OAAc,gBAAd,C;UAAc,2B;UACV,IAAI,gBAAW,UAAK,OAAL,CAAX,CAAJ,C;YACI,OAAAO,O;;;;MAInB,OAAO,E;K;IAGX,2C;MAIkB,Q;MAAA,OAAQ,WAAR,wBAAAQ,CAAR,W;MAAd,OAAc,cAAd,C;QAAc,uB;QACV,IAAI,YAAW,UAAK,KAAL,CAAf,C;UACI,OAAO,K;;;MAGf

OAAO,kBAAO,cAAP,C;O;KARX,C;oFAWA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;O;KARX,C;oFAWA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;O;KARX,C;oFAWA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;O;KARX,C;oFAWA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;O;KARX,C;IAWA,qC;MAOI,IAoxKO,qBAAQ,CApxKf,C;QACI,MAAM,2BAAuB,iBAAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;K;IAGX,qC;MAOI,IAgxKO,qBAAQ,CAhxKf,C;QACI,MAAM,2BAAuB,iBAAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;K;IAGX,sC;MAOI,IA4wKO,qBAAQ,CA5wKf,C;QACI,MAAM,2BAAuB,iBAAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;K;IAGX,sC;MAOI,IAwwKO,qBAAQ,CAxwKf,C;QACI,MAAM,2BAAuB,iBAAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;K;IAGX,sC;MAOI,IAowKO,qBAAQ,CApwKf,C;QACI,MAAM,2BAAuB,iBAAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;K;IAGX,sC;MAOI,IAgwKO,qBAAQ,CAhwKf,C;QACI,MAAM,2BAAuB,iBAAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;K;IAGX,sC;MAOI,IA4vKO,qBAAQ,CA5vKf,C;QACI,MAAM,2BAAuB,iBAAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;K;IAGX,sC;MAOI,IAwvKO,qBAAQ,CAxvKf,C;QACI,MAAM,2BAAuB,iBAAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;K;IAGX,sC;MAOI,IAovKO,qBAAQ,CApvKf,C;QACI,MAAM,2BAAuB,iBAAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;K;8FAGX,yB;MAAA,mC;MAAA,4D;MAAA,4B;QAOI,OAAO,wBAAa,cAAAb,C;O;KAPxB,C;gGAUA

B;MAAA,uC;QAMoB,UAST,M;QAXP,aAAiB,I;QACjB,YAAY,K;QACZ,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,gDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAAY,MAAM,gCAAuB,mDAAvB,C;QAElB,OAAO,6E;O;KAfX,C;oFAkBA,yB;MAAA,kF;MAAA,iE;MAAA,8B;MAAA,uC;QAMoB,UAST,M;QAXP,aAAoB,I;QACpB,YAAY,K;QACZ,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,gDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAAY,MAAM,gCAAuB,mDAAvB,C;QAElB,OAAO,2D;O;KAfX,C;qFAkBA,yB;MAAA,kF;MAAA,iE;MAAA,8B;MAAA,uC;QAMoB,UAST,M;QAXP,aAAqB,I;QACrB,YAAY,K;QACZ,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,gDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAAY,MAAM,gCAAuB,mDAAvB,C;QAElB,OAAO,2D;O;KAfX,C;qFAkBA,yB;MAAA,kF;MAAA,iE;MAAA,8B;MAAA,uC;QAMoB,UAST,M;QAXP,aAAqB,I;QACrB,YAAY,K;QACZ,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,gDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAAY,MAAM,gCAAuB,mDAAvB,C;QAElB,OAAO,iE;O;KAfX,C;qFAkBA,yB;MAAA,kF;MAAA,iE;MAAA,8B;MAAA,uC;QAMoB,UAST,M;QAXP,aAAqB,I;QACrB,YAAY,K;QACZ,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,gDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAAY,MAAM,gCAAuB,mDAAvB,C;QAElB,OAAO,2D;O;KAfX,C;qFAkBA,yB;MAAA,kF;MAAA,iE;MAAA,8B;MAAA,uC;QAMoB,UAST,M;QAXP,aAAqB,I;QACrB,YAAY,K;QACZ,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,gDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAAY,MAAM,gCAAuB,mDAAvB,C;QAElB,OAAO,iE

O,I;UAClB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;K;iGA
GX,gC;MAMoB,Q;MAFhB,aAAoB,I;MACpB,YAAY,K;MACZ,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;
QACI,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAAW,OAAO,I;UAClB,SAAS,O;UACT,QAAQ,I;;;M
AGhB,IAAI,CAAC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;K;iGAGX,gC;MAMoB,Q;MAFhB,aAAqB,I;M
ACrB,YAAY,K;MACZ,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,IAAI,UAAU,OAAV,CAAJ,C;UA
CI,IAAI,KAAJ,C;YAAW,OAAO,I;UAClB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAAY,OA
AO,I;MACnB,OAAO,M;K;iGAGX,gC;MAMoB,Q;MAFhB,aAAsB,I;MACtB,YAAY,K;MACZ,wBAAgB,SAAhB
,gB;QAAgB,cAAA,SAAhB,M;QACI,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAAW,OAAO,I;UAClB
,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;K;iGAGX,gC;M
AMoB,Q;MAFhB,aAAuB,I;MACvB,YAAY,K;MACZ,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,IA
AI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAAW,OAAO,I;UAClB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IA
AI,CAAC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;K;iGAGX,yB;MAAA,oC;MAAA,gC;MAAA,uC;QAMo
B,Q;QAFhB,aAAoB,I;QACpB,YAAY,K;QACZ,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UACI,
IAAI,UAAU,oBAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,OAAO,I;YAClB,SAAS,O;YACT,QAAQ,I;;;QAGhB,I
AAI,CAAC,KAAL,C;UAAY,OAAO,I;QACnB,OAAO,M;O;KAdX,C;IAiBA,4B;McvqGI,IAAI,Ed+qGI,KAAK,Cc
/qGT,CAAJ,C;QACI,cd8qGc,sD;Qc7qGd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Md8qGV,OAAO,oBAAoB,gBAA
V,mBAAO,CAAP,IAAU,EAAc,CAAd,CAApB,C;K;IAGX,8B;McnrGI,IAAI,Ed2rGI,KAAK,Cc3rGT,CAAJ,C;QA
CI,cd0rGc,sD;QczrGd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Md0rGV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,
IAAU,EAAc,CAAd,CAApB,C;K;IAGX,8B;Mc/rGI,IAAI,EdusGI,KAAK,CcvsGT,CAAJ,C;QACI,cdssGc,sD;Qcrs
Gd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MdssGV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd,
CApB,C;K;IAGX,8B;Mc3sGI,IAAI,EdmtGI,KAAK,CcntGT,CAAJ,C;QACI,cdktGc,sD;QcjtGd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MdktGV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd,CAApB,C;K;IAGX,
8B;McvtGI,IAAI,Ed+tGI,KAAK,Cc/tGT,CAAJ,C;QACI,cd8tGc,sD;Qc7tGd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;
Md8tGV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd,CAApB,C;K;IAGX,8B;McnuGI,IAAI,Ed
2uGI,KAAK,Cc3uGT,CAAJ,C;QACI,cd0uGc,sD;QczuGd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Md0uGV,OAAO,
sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd,CAApB,C;K;IAGX,8B;Mc/uGI,IAAI,EduvGI,KAAK,CcvvG
GT,CAAJ,C;QACI,cdsvGc,sD;QcrvGd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MdsvGV,OAAO,sBAAoB,gBAAV,
mBAAO,CAAP,IAAU,EAAc,CAAd,CAApB,C;K;IAGX,8B;Mc3vGI,IAAI,EdmwGI,KAAK,CcnwGT,CAAJ,C;QACI,Q
dkwGc,sD;QcjwGd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MdkwGV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,CA
AP,IAAU,EAAc,CAAd,CAApB,C;K;IAGX,8B;McvwGI,IAAI,Ed+wGI,KAAK,Cc/wGT,CAAJ,C;QACI,cd8wGc,s
D;Qc7wGd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Md8wGV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EA
Ac,CAAd,CApB,C;K;IAGX,gC;McnxGI,IAAI,Ed2xGI,KAAK,Cc3xGT,CAAJ,C;QACI,cd0xGc,sD;QczxGd,MA
AM,gCAAyB,OAAQ,WAAjC,C,C;;Md0xGV,OAAO,gBAAgB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd,CAAh
B,C;K;IAGX,kC;Mc/xGI,IAAI,EduyGI,KAAK,CcvyGT,CAAJ,C;QACI,cdsyGc,sD;QcryGd,MAAM,gCAAyB,OA
AQ,WAAjC,C,C;;MdsyGV,OAAO,kBAAgB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,kC;
Mc3yGI,IAAI,EdmzGI,KAAK,CcnzGT,CAAJ,C;QACI,cdkzGc,sD;QcjzGd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;
MdkzGV,OAAO,kBAAgB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,kC;McvzGI,IAAI,Ed
+zGI,KAAK,Cc/zGT,CAAJ,C;QACI,cd8zGc,sD;Qc7zGd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Md8zGV,OAAO,k
BAAgB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,kC;Mcn0GI,IAAI,Ed20GI,KAAK,Cc30
GT,CAAJ,C;QACI,cd00Gc,sD;Qcz0Gd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Md00GV,OAAO,kBAAgB,gBAAV,
mBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,kC;Mc/0GI,IAAI,Edu1GI,KAAK,Ccv1GT,CAAJ,C;QAC
I,cds1Gc,sD;Qcr1Gd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Mds1GV,OAAO,kBAAgB,gBAAV,mBAAO,CAAP,IA
AU,EAAc,CAAd,CAAhB,C;K;IAGX,kC;Mc31GI,IAAI,Edm2GI,KAAK,Ccn2GT,CAAJ,C;QACI,cdk2Gc,sD;Qcj2
Gd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Mdk2GV,OAAO,kBAAgB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd
,CAAhB,C;K;IAGX,kC;Mcv2GI,IAAI,Ed+2GI,KAAK,Cc/2GT,CAAJ,C;QACI,cd82Gc,sD;Qc72Gd,MAAM,gCA
AyB,OAAQ,WAAjC,C,C;;Md82GV,OAAO,kBAAgB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IA
GX,kC;Mcn3GI,IAAI,Ed23GI,KAAK,Cc33GT,CAAJ,C;QACI,cd03Gc,sD;Qcz3Gd,MAAM,gCAAyB,OAAQ,WA
AjC,C,C;;Md03GV,OAAO,kBAAgB,gBAAV,mBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;gGAGX,yB;MAAA

,8D;MAAA,4C;MAAA,qD;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,W;O;KAXX,C;kGAcA,yB;MAAA,8D;MAAA,2C;MAAA,qD;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,W;O;KAXX,C;kGAcA,yB;MAAA,8D;MAAA,4

AAhB,e;UAAsB,IArgBc,SAqgBV,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QArgB1D,OAsgBO,W;O;KA5gBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAsgBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAtgBY,SAsgBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAtgB1D,OAugBO,W;O;KA7gBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAugBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAvgBY,SAugBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAvgB1D,OAwgBO,W;O;KA9gBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAwgBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAxgBY,SAwgBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAxgB1D,OAygBO,W;O;KA/gBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAygBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAzgBY,SAygBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAzgB1D,OA0gBO,W;O;KAhhBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA0gBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA1gBY,SA0gBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA1gB1D,OA2gBO,W;O;KAjhBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA2gBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA3gBY,SA2gBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA3gB1D,OA4gBO,W;O;KAlhBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA4gBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA5gBY,SA4gBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA5gB1D,OA6gBO,W;O;KAnhBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA6gBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA7gBY,SA6gBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA7gB1D,OA8gBO,W;O;KAphBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA8gBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA9gBY,SA8gBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA9gB1D,OA+gBO,W;O;KArhBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA+gBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA/gBY,SA+gBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA/gB1D,OAghBO,W;O;KAthBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAghBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAhhBY,SAghBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAhhB1D,OAihBO,W;O;KAvhBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAihBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAjhBY,SAihBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAjhB1D,OAkhBO,W;O;KAxhBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAkhBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAlhBY,SAkhBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAlhB1D,OAmhBO,W;O;KAzhBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAmhBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAnhBY,SAmhBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAnhB1D,OAohBO,W;O;KA1hBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAohBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAphBY,SAohBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAphB1D,OAqhBO,W;O;KA3hBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAqhBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IArhBY,SAqhBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QArhB1D,OAshBO,W;O;KA5hBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAshBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAthBY,SAshBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAthB1D,OAuhBO,W;O;KA7hBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAuhBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAvhBY,SAuhBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAvhB1D,OAwhBO,W;O;KA9hBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAwhBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAxhBY,SAwhBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAxhB1D,OAyhBO,W;O;KA/hBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAyhBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAzhBY,SAyhBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAzhB1D,OA0hBO,W;O;KAhiBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA0hBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA1hBY,SA0hBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA1hB1D,OA2hBO,W;O;KAjiBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA2hBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA3hBY,SA2hBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA3hB1D,OA4hBO,W;O;KAliBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA4hBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA5hBY,SA4hBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA5hB1D,OA6hBO,W;O;KAniBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA6hBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA7hBY,SA6hBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA7hB1D,OA8hBO,W;O;KApiBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA8hBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA9hBY,SA8hBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA9hB1D,OA+hBO,W;O;KAriBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA+hBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA/hBY,SA+hBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA/hB1D,OAgiBO,W;O;KAtiBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAgiBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAhiBY,SAgiBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAhiB1D,OAiiBO,W;O;KAviBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAiiBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAjiBY,SAiiBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAjiB1D,OAkiBO,W;O;KAxiBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAkiBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAliBY,SAkiBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAliB1D,OAmiBO,W;O;KAziBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAmiBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAniBY,SAmiBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAniB1D,OAoiBO,W;O;KA1iBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAoiBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IApiBY,SAoiBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QApiB1D,OAqiBO,W;O;KA3iBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAqiBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAriBY,SAqiBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAriB1D,OAsiBO,W;O;KA5iBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAsiBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAtiBY,SAsiBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAtiB1D,OAuiBO,W;O;KA7iBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAuiBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAviBY,SAuiBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAviB1D,OAwiBO,W;O;KA9iBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAwiBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAxiBY,SAwiBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAxiB1D,OAyiBO,W;O;KA/iBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAyiBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAziBY,SAyiBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAziB1D,OA0iBO,W;O;KAhjBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA0iBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA1iBY,SA0iBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA1iB1D,OA2iBO,W;O;KAjjBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA2iBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA3iBY,SA2iBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA3iB1D,OA4iBO,W;O;KAljBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA4iBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA5iBY,SA4iBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA5iB1D,OA6iBO,W;O;KAnjBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA6iBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA7iBY,SA6iBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA7iB1D,OA8iBO,W;O;KApjBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA8iBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA9iBY,SA8iBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA9iB1D,OA+iBO,W;O;KArjBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA+iBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA/iBY,SA+iBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA/iB1D,OAgjBO,W;O;KAtjBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAgjBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAhjBY,SAgjBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAhjB1D,OAijBO,W;O;KAvjBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAijBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAjjBY,SAijBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAjjB1D,OAkjBO,W;O;KAxjBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAkjBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAljBY,SAkjBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAljB1D,OAmjBO,W;O;KAzjBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAmjBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAnjBY,SAmjBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAnjB1D,OAojBO,W;O;KA1jBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAojBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IApjBY,SAojBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QApjB1D,OAqjBO,W;O;KA3jBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAqjBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IArjBY,SAqjBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QArjB1D,OAsjBO,W;O;KA5jBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAsjBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAtjBY,SAsjBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAtjB1D,OAujBO,W;O;KA7jBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAujBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAvjBY,SAujBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAvjB1D,OAwjBO,W;O;KA9jBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAwjBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAxjBY,SAwjBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAxjB1D,OAyjBO,W;O;KA/jBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAyjBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAzjBY,SAyjBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAzjB1D,OA0jBO,W;O;KAhkBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA0jBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA1jBY,SA0jBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA1jB1D,OA2jBO,W;O;KAjkBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA2jBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA3jBY,SA2jBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA3jB1D,OA4jBO,W;O;KAlkBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA4jBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA5jBY,SA4jBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA5jB1D,OA6jBO,W;O;KAnkBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA6jBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA7jBY,SA6jBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA7jB1D,OA8jBO,W;O;KApkBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA8jBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA9jBY,SA8jBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA9jB1D,OA+jBO,W;O;KArkBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA+jBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA/jBY,SA+jBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA/jB1D,OAgkBO,W;O;KAtkBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAgkBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAhkBY,SAgkBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAhkB1D,OAikBO,W;O;KAvkBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAikBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAjkBY,SAikBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAjkB1D,OAkkBO,W;O;KAxkBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAkkBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAlkBY,SAkkBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAlkB1D,OAmkBO,W;O;KAzkBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAmkBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAnkBY,SAmkBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAnkB1D,OAokBO,W;O;KA1kBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAokBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IApkBY,SAokBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QApkB1D,OAqkBO,W;O;KA3kBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAqkBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IArkBY,SAqkBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QArkB1D,OAskBO,W;O;KA5kBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAskBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAtkBY,SAskBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAtkB1D,OAukBO,W;O;KA7kBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAukBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAvkBY,SAukBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAvkB1D,OAwkBO,W;O;KA9kBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAwkBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAxkBY,SAwkBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAxkB1D,OAykBO,W;O;KA/kBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAykBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAzkBY,SAykBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAzkB1D,OA0kBO,W;O;KAhlBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA0kBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA1kBY,SA0kBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA1kB1D,OA2kBO,W;O;KAjlBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA2kBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA3kBY,SA2kBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA3kB1D,OA4kBO,W;O;KAllBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA4kBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA5kBY,SA4kBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA5kB1D,OA6kBO,W;O;KAnlBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA6kBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA7kBY,SA6kBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA7kB1D,OA8kBO,W;O;KAplBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA8kBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA9kBY,SA8kBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA9kB1D,OA+kBO,W;O;KArlBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA+kBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA/kBY,SA+kBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA/kB1D,OAglBO,W;O;KAtlBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAglBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAhlBY,SAglBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAhlB1D,OAilBO,W;O;KAvlBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAilBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAjlBY,SAilBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAjlB1D,OAklBO,W;O;KAxlBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAklBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAllBY,SAklBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAllB1D,OAmlBO,W;O;KAzlBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAmlBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAnlBY,SAmlBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAnlB1D,OAolBO,W;O;KA1lBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAolBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAplBY,SAolBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAplB1D,OAqlBO,W;O;KA3lBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAqlBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IArlBY,SAqlBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QArlB1D,OAslBO,W;O;KA5lBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAslBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAtlBY,SAslBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAtlB1D,OAulBO,W;O;KA7lBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAulBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAvlBY,SAulBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAvlB1D,OAwlBO,W;O;KA9lBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAwlBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAxlBY,SAwlBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAxlB1D,OAylBO,W;O;KA/lBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAylBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAzlBY,SAylBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAzlB1D,OA0lBO,W;O;KAhmBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA0lBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA1lBY,SA0lBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA1lB1D,OA2lBO,W;O;KAjmBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA2lBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA3lBY,SA2lBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA3lB1D,OA4lBO,W;O;KAlmBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA4lBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA5lBY,SA4lBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA5lB1D,OA6lBO,W;O;KAnmBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA6lBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA7lBY,SA6lBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA7lB1D,OA8lBO,W;O;KApmBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA8lBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA9lBY,SA8lBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA9lB1D,OA+lBO,W;O;KArmBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QA+lBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IA/lBY,SA+lBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QA/lB1D,OAgmBO,W;O;KAtmBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAgmBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAhmBY,SAgmBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAhmB1D,OAimBO,W;O;KAvmBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAimBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAjmBY,SAimBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAjmB1D,OAkmBO,W;O;KAxmBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAkmBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAlmBY,SAkmBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAlmB1D,OAmmBO,W;O;KAzmBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAmmBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAnmBY,SAmmBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAnmB1D,OAomBO,W;O;KA1mBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAomBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IApmBY,SAomBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QApmB1D,OAqmBO,W;O;KA3mBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAqmBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IArmBY,SAqmBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QArmB1D,OAsmBO,W;O;KA5mBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAsmBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAtmBY,SAsmBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAtmB1D,OAumBO,W;O;KA7mBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAumBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAvmBY,SAumBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAvmB1D,OAwmBO,W;O;KA9mBX,C;oFASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAS,gB;QAwmBA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAxmBY,SAwmBR,CAAU,OAAV,CAAJ,C;YAAwB,WAAY,WAAI,OAAJ,C;;QAxmB1D,OAym

A14S3B,CAAJ,C;UAA2C,sBA04SZ,IA14SY,C;;MAE/C,OAAO,W;K;sGAGX,6C;MA+4SiB,gB;MADb,YAAY,C;
MACZ,iD;QAAa,WAAb,e;QAt4SI,IAAI,WAs4SkB,cAt4SlB,EAs4SkB,sBAt4SlB,WAs4S2B,IAt4S3B,CAAJ,C;U
AA2C,sBAs4SZ,IAt4SY,C;;MAE/C,OAAO,W;K;sGAGX,yB;MAAA,gC;MAo4SA,oC;MAp4SA,oD;QA24SiB,gB
;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,0B;UAAmB,eAAO,cAAP,EAAO,sBAAP,S;UAAA,cAAgB,iB;UAl4S/
B,IAAI,UAAU,OAAV,EAAiB,OAAjB,CAAJ,C;YAA2C,sBAAI,OAAJ,C;;QAE/C,OAAO,W;O;KAXX,C;sGAcA,
yB;MAAA,+D;MAAA,sC;QAMW,kBAAmB,gB;QASV,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAI,YAAJ,C;
YAAkB,WAAY,WAAI,OAAJ,C;;QATpD,OAUO,W;O;KAhBX,C;0GASA,4C;MAMoB,Q;MAAhB,wBAAgB,SA
AhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,YAAJ,C;UAAkB,WAAY,WAAI,OAAJ,C;;MACpD,OAAO,W;
K;wFAGX,yB;MAAA,+D;MAAA,uC;QAMW,kBAAY,gB;QAoGH,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAA
I,CApGS,SAoGR,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QApG3D,OAqGO,W;O;KA3GX,C;0
FASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAY,gB;QAqGH,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAI,CA
rGY,SAqGX,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QArG3D,OAsGO,W;O;KA5GX,C;0FAS
A,yB;MAAA,+D;MAAA,uC;QAMW,kBAAY,gB;QAsGH,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAI,CAtGa,
SAsGZ,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAtG3D,OAuGO,W;O;KA7GX,C;0FASA,yB;
MAAA,+D;MAAA,uC;QAMW,kBAAY,gB;QAuGH,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAI,CAvGW,SAu
GV,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAvG3D,OAwGO,W;O;KA9GX,C;0FASA,yB;M
AAA,+D;MAAA,uC;QAMW,kBAAY,gB;QAwGH,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAI,CAxGY,SAwG
X,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAxG3D,OAyGO,W;O;KA/GX,C;0FASA,yB;MAA
A,+D;MAAA,uC;QAMW,kBAAY,gB;QAyGH,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAI,CAzGa,SAyGZ,CA
AU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAzG3D,OA0GO,W;O;KAhHX,C;0FASA,yB;MAAA,+D
;MAAA,uC;QAMW,kBAAY,gB;QA0GH,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAI,CA1Gc,SA0Gb,CAAU,
OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QA1G3D,OA2GO,W;O;KAjHX,C;0FASA,yB;MAAA,+D;MA
AA,uC;QAMW,kBAAY,gB;QA2GH,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAAI,CA3Ge,SA2Gd,CAAU,OAA
V,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QA3G3D,OA4GO,W;O;KAlHX,C;0FASA,yB;MAAA,+D;MA4GA,
oC;MAAA,gC;MA5GA,uC;QAMW,kBAAY,gB;QA4GH,Q;QAAhB,iD;UAAgB,cAAhB,0B;UAAsB,IAAI,CA5G
Y,SA4GX,CAAU,oBAAV,CAAL,C;YAAyB,WAAY,WAAI,oBAAJ,C;;QA5G3D,OA6GO,W;O;KAnHX,C;IASA,
kC;MAMI,OAAO,2BAAgB,gBAAhB,C;K;IAGX,iD;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,S
AAhB,M;QAAsB,IAAI,eAAJ,C;UAAqB,WAAY,WAAI,OAAJ,C;;MACvD,OAAO,W;K;4FAGX,6C;MAMoB,Q;
MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,
WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;K;8FAGX,6C;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,c
AAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO
,W;K;8FAGX,6C;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UA
AU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;K;8FAGX,6C;MAMoB,Q;MAAhB,wB
AAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WA
AI,OAAJ,C;;MAC3D,OAAO,W;K;8FAGX,6C;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB
,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;K;8FAGX
,6C;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,C
AAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;K;8FAGX,6C;MAMoB,Q;MAAhB,wBAAgB,SAAh
B,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;
MAC3D,OAAO,W;K;8FAGX,6C;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,I
AAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;K;8FAGX,yB;MAAA,
oC;MAAA,gC;MAAA,oD;QAMoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UAAsB
,IAAI,CAAC,UAAU,oBAAV,CAAL,C;YAAyB,WAAY,WAAI,oBAAJ,C;;QAC3D,OAAO,W;O;KAPX,C;sFAUA
,6C;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;
UAAwB,WAAY,WAAI,OAAJ,C;;MAC1D,OAAO,W;K;wFAGX,6C;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;
QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,WAAY,WAAI,OAAJ,C;;MAC1D,OA
AO,W;K;wFAGX,6C;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU
,OAAV,CAAJ,C;UAAwB,WAAY,WAAI,OAAJ,C;;MAC1D,OAAO,W;K;wFAGX,6C;MAMoB,Q;MAAhB,wBA

AgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,WAAY,WAAI,OAAJ,C;;MAC1D,OAAO,W;K;wFAGX,6C;MAMoB,Q;MAAhB,wBAA

ACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QAChB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAAK,WAAL
,C;;MAE5B,OAAO,M;K;IAGX,0C;MAMwB,UACT,M;MAHX,aAAa,eAAW,OAAQ,KAAnB,C;MACb,kBAAkB,
C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QAChB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAAK,W
AAL,C;;MAE5B,OAAO,M;K;IAGX,0C;MAMwB,UACT,M;MAHX,aAAa,eAAS,OAAQ,KAAjB,C;MACb,kBAA
kB,C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QAChB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAAK
,WAAL,C;;MAE5B,OAAO,M;K;IAGX,0C;MAMwB,UACT,M;MAHX,aAAa,iBAAU,OAAQ,KAAlB,C;MACb,k
BAAkB,C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QAChB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,U
AAK,WAAL,C;;MAE5B,OAAO,M;K;IAGX,0C;MAMwB,UACT,M;MAHX,aAAa,iBAAW,OAAQ,KAAnB,C;M
ACb,kBAAkB,C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QAChB,OAAO,oBAAP,EAAO,4BAAP,YA
AwB,UAAK,WAAL,C;;MAE5B,OAAO,M;K;IAGX,0C;MAMwB,UACT,M;MAHX,aAAa,iBAAY,OAAQ,KAAp
B,C;MACb,kBAAkB,C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QAChB,OAAO,oBAAP,EAAO,4BAA
P,YAAwB,UAAK,WAAL,C;;MAE5B,OAAO,M;K;IAGX,0C;MAMwB,UACT,M;MAHX,aAAa,oBAAa,oBAAQ,K
AArB,C;MACb,kBAAkB,C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QAChB,OAAO,oBAAP,EAAO,4
BAAP,YAAwB,UAAK,WAAL,C;;MAE5B,OAAO,M;K;IAGX,0C;MAMwB,UACT,M;MAHX,aAAa,iBAAU,OAQ,KAAlB,C;MACb,kBAAkB,C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QAChB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAAK,WAAL,C;;MAE5B,OAAO,M;K;IAGX,0C;MAII,IAAI,OAAQ,UAAZ,C;QAAuB,OA
AO,yBAAY,CAAZ,EAAe,CAAf,C;MAC9B,OAAO,yBAAY,OAAQ,MAApB,EAA2B,OAAQ,aAAR,GAAuB,CA
AvB,IAA3B,C;K;IAGX,0C;MAII,IAAI,OAAQ,UAAZ,C;QAAuB,OAAO,cAAU,CAAV,C;MAC9B,OAAO,yBAA
Y,OAAQ,MAApB,EAA2B,OAAQ,aAAR,GAAuB,CAAvB,IAA3B,C;K;IAGX,2C;MAII,IAAI,OAAQ,UAAZ,C;Q
AAuB,OAAO,eAAW,CAAX,C;MAC9B,OAAO,yBAAY,OAAQ,MAApB,EAA2B,OAAQ,aAAR,GAAuB,CAAvB,
,IA3B,C;K;IAGX,2C;MAII,IAAI,OAAQ,UAAZ,C;QAAuB,OAAO,eAAS,CAAT,C;MAC9B,OAAO,yBAAY,OA
AQ,MAApB,EAA2B,OAAQ,aAAR,GAAuB,CAAvB,IAA3B,C;K;IAGX,2C;MAII,IAAI,OAAQ,UAAZ,C;QAAuB
,OAAO,iBAAU,CAAV,C;MAC9B,OAAO,yBAAY,OAAQ,MAApB,EAA2B,OAAQ,aAAR,GAAuB,CAAvB,IA
3B,C;K;IAGX,2C;MAII,IAAI,OAAQ,UAAZ,C;QAAuB,OAAO,iBAAW,CAAX,C;MAC9B,OAAO,yBAAY,OAA
Q,MApB,EA2B,OAAQ,aAAR,GAAuB,CAAvB,IA3B,C;K;IAGX,2C;MAII,IAAI,OAAQ,UAAZ,C;QAAuB,
OAAO,iBAAY,CAAZ,C;MAC9B,OAAO,yBAAY,OAAQ,MAApB,EA2B,OAAQ,aAAR,GAAuB,CAAvB,IA3
B,C;K;IAGX,2C;MAII,IAAI,OAAQ,UAAZ,C;QAAuB,OAAO,oBAAa,oBAAQ,CAAR,C;MAC9B,0BAAY,
OAAQ,MAApB,EA2B,OAAQ,aAAR,GAAuB,CAAvB,IA3B,C;K;IAGX,4B;MAciB,Q;Mc3nJb,IAAI,EdqnJI,KAAK,CcrnJT,CAAJ,C;QACI,cdonJc,sD;QcnnJd,MAAM,gCAAyB,
OAAQ,WAAjC,C,C;;MdonJV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,iB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,
WAAW,iBAAa,CAAb,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAK,WAAI,IAAJ,C;QACL,I
AAI,mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,8B;MAciB,Q;McjpJb,IAAI,Ed2oJI,KAAK,Cc3oJT,CA
AJ,C;QACI,cd0oJc,sD;QczoJd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Md0oJV,IAAI,MAAK,CAAT,C;QAAY,OAA
O,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,U
AAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,
WAAA,SAAb,M;QACI,IAAK,WAAI,IAAJ,C;QACL,IAAI,mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,
8B;MAciB,Q;McvqJb,IAAI,EdiqJI,KAAK,CcjqJT,CAAJ,C;QACI,cdgqJc,sD;Qc/pJd,MAAM,gCAAyB,
OAAQ,WAAjC,C,C;;MdgqJV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,mB;
MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW
,iBAAiB,CAAjB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,WAAA,SAAb,M;QACI,IAAK,WAAI,IAAJ,C;QACL,IAAI,
mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,8B;MAciB,Q;Mc7rJb,IAAI,EdurJI,KAAK,CcvrJT,CAAJ,C;
QACI,cdsrJc,sD;QcrrJd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MdsrJV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;M
ACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,C
AAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAe,CAAf,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,WAAA,SAA
b,M;QACI,IAAK,WAAI,IAAJ,C;QACL,IAAI,mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,8B;MAciB,Q
;McntJb,IAAI,Ed6sJI,KAAK,Cc7sJT,CAAJ,C;QACI,cd4sJc,sD;Qc3sJd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Md4

sJV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAK,WAAI,IAAJ,C;QACL,IAAI,mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,8B;MACiB,Q;MczuJb,IAAI,EdmuJI,KAAK,CcnuJT,CAAJ,C;QACI,cdkuJc,sD;QcjuJd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MdkuJV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAK,WAAI,IAAJ,C;QACL,IAAI,mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,8B;MACiB,Q;Mc/vJb,IAAI,EdyvJI,KAAK,CczuJT,CAAJ,C;QACI,cdwvJc,sD;QcvvJd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MdwvJV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAK,WAAI,IAAJ,C;QACL,IAAI,mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,8B;McrxJb,IAAI,Ed+wJI,KAAK,Cc/wJT,CAAJ,C;QACI,cd8wJc,sD;Qc7wJd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;Md8wJV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAK,WAAI,IAAJ,C;QACL,IAAI,mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,8B;Mc3yJb,IAAI,EdqyJI,KAAK,CcryJT,CAAJ,C;QACI,cdoyJc,sD;QcnyJd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MdoyJV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAK,WAAI,IAAJ,C;QACL,IAAI,mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,8B;McjuJb,IAAI,EdquJI,KAAK,CcruJT,CAAJ,C;QACI,cdouJc,sD;QcnuJd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MdouJV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAK,WAAI,IAAJ,C;QACL,IAAI,mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,8B;Mc7yJb,IAAI,EdqyJI,KAAK,CcryJT,CAAJ,C;QACI,cdoyJc,sD;QcnyJd,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MdoyJV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAK,WAAI,IAAJ,C;QACL,IAAI,mCAAW,CAAf,C;UACI,K;;MAER,OAAO,I;K;IAGX,8B;Mc3

AyB,OAAQ,WAAjC,C;;Md46JV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,gB;MACX,IAAI,KAAK,IAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,OAAO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAkB,CAAlB,C;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,UAAK,KAAL,CAAJ,C;MACT,OAAO,I;K;IAGX,kC;Mcx7JI,IAAI,Edg8JI,KAAK,Cch8JT,CAAJ,C;QACI,cd+7Jc,sD;Qc97Jd,MAAM,gCAAyB,OAAAQ,WAAjC,C;;Md+7JV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,gB;MACX,IAAI,KAAK,IAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,OAAO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAkB,CAAlB,mB,CAAqB,C;MAChC,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,UAAK,KAAL,CAAJ,C;MACT,OAAO,I;K;IAGX,kC;Mc38JI,IAAI,Edm9JI,KAAK,Ccn9JT,CAAJ,C;QACI,cdk9Jc,sD;Qcj9Jd,MAAM,gCAAyB,OAAAQ,WAAjC,C;;Mdk9JV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,gB;MACX,IAAI,KAAK,IAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,OAAO,CAAP,IAAL,EAAP,C;MACnB,WAAW,iBAAkB,gB,CAAlB,C;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,sBAAK,KAAL,CAAJ,C;MACT,OAAO,I;K;gBAGX,kB;MAAA,8D;MAAA,4C;MAAA,gD;MAAA,uC;QAAM,IAAI,iBAAc,wBAAd,WAA+B,C,AAA/B,U;QAAU,IAAI,IAAJ,UAAS,IAAT,UAAc,IAAd,UAAmB,IAAnB,UAAwB,IAAxB,UAA6B,IAA7B,UAAkC,IAAlC,UAAuC,IAAvC,UAA4C,IAA5C,UAAiD,IAAjD,UAAsD,IAAtD,UAA2D,IAA3D,UAAgE,IAAhE,UAAqE,IAArE,UAA0E,IAA1E,UAA+E,IAA/E,UAAoF,IAApF,UAAyF,IAAzF,UAA8F,IAA9F,UAAmG,IAAnG,UAAwG,IAAxG,UAA6G,IAA7G,UAAkH,IAAlH,UAAuH,IAAvH,UAA4H,IAA5H,UAAiI,IAAjI,UAAsI,IAAtI,UAA2I,IAA3I,UAAgJ,IAAhJ,UAAqJ,IAArJ,UAA0J,IAA1J,UAA+J,IAA/J,UAAoK,IAApK,UAAyK,IAAzK,UAA8K,IAA9K,UAAmL,IAAnL,UAAwL,IAAxL,UAA6L,IAA7L,UAAkM,IAAlM,UAAuM,IAAvM,UAA4M,IAA5M,UAAiN,IAAjN,UAAsN,IAAtN,UAA2N,IAA3N,UAAgO,IAAhO,UAAqO,IAArO,UAA0O,IAA1O,UAA+O,IAA/O,UAAoP,IAApP,UAAyP,IAAzP,UAA8P,IAA9P,UAAmQ,IAAnQ,UAAwQ,IAAxQ,UAA6Q,IAA7Q,UAAkR,IAAlR,UAAuR,IAAvR,UAA4R,IAA5R,UAAiS,IAAjS,UAAsS,IAAtS,UAA2S,IAA3S,UAAgT,IAAhT,UAAqT,IAArT,UAA0T,IAA1T,UAA+T,IAA/T,UAAoU,IAApU,UAAyU,IAAzU,UAA8U,IAA9U,UAAmV,IAAnV,UAAwV,IAAxV,UAA6V,IAA7V,UAAkW,IAAlW,UAAuW,IAAvW,UAA4W,IAA5W,UAAiX,IAAjX,UAAsX,IAAtX,UAA2X,IAA3X,UAAgY,IAAhY,UAAqY,IAArY,UAA0Y,IAA1Y,UAA+Y,IAA/Y,UAAoZ,IAApZ,UAAyZ,IAAzZ,UAA8Z,IAA9Z,UAAma,IAAna,UAAwa,IAAxa,UAA6a,IAA7a,UAAkb,IAAlb,UAAub,IAAvb,UAA4b,IAA5b,UAAic,IAAjc,UAAsc,IAAtc,UAA2c,IAA3c,UAAgd,IAAhd,UAAqd,IAArd,UAA0d,IAA1d,UAA+d,IAA/d,UAAoe,IAApe,UAAyeAAAA,IAAze,UAA8e,IAA9e,UAAmf,IAAnf,UAAwf,IAAxf,UAA6f,IAA7f,UAAkgB,IAAlgB,UAAugB,IAAvgB,UAA4gB,IAA5gB,UAAihB,IAAjhB,UAAshB,IAAthB,UAA2hB,IAA3hB,UAAgiB,IAAhiB,UAAqiB,IAAriB,UAA0iB,IAA1iB,UAA+iB,IAA/iB,UAAojB,IAApjB,UAAyjB,IAAzjB,UAA8jB,IAA9jB,UAAmkB,IAAnkB,UAAwkB,IAAxkB,UAA6kB,IAA7kB,UAAklB,IAAllB,UAAulB,IAAvlB,UAA4lB,IAA5lB,UAAimB,IAAjmB,UAAsmB,IAAtmB,UAA2mB,IAA3mB,UAAgnB,IAAhnB,UAAqnB,IAArnB,UAA0nB,IAA1nB,UAA+nB,IAA/nB,UAAooB,IAApoB,UAAyoB,IAAzoB,UAA8oB,IAA9oB,UAAmpB,IAAnpB,UAAwpB,IAAxpB,UAA6pB,IAA7pB,UAAkqB,IAAlqB,UAAuqB,IAAvqB,UAA4qB,IAA5qB,UAAirB,IAAjrB,UAAsrB,IAAtrB,UAA2rB,IAA3rB,UAAgsB,IAAhsB,UAAqsB,IAArsB,UAA0sB,IAA1sB,UAA+sB,IAA/sB,UAAotB,IAAptB,UAAytB,IAAztB,UAA8tB,IAA9tB,UAAmuB,IAAnuB,UAAwuB,IAAxuB,UAA6uB,IAA7uB,UAAkvB,IAAlvB,UAAuvB,IAAvvB,UAA4vB,IAA5vB,UAAiwB,IAAjwB,UAAswB,IAAtwB,UAA2wB,IAA3wB,UAAgxB,IAAhxB,UAAqxB,IAArxB,UAA0xB,IAA1xB,UAA+xB,IAA/xB,UAAoyB,IAApyB,UAAyyB,IAAzyB,UAA8yB,IAA9yB,UAAmzB,IAAnzB,UAAwzB,IAAxzB,UAA6zB,IAA7zB,UAAk0B,IAAl0B,UAAu0B,IAAv0B,UAA40B,IAA50B,UAAi1B,IAAj1B,UAAs1B,IAAt1B,UAA21B,IAA31B,UAAg2B,IAAh2B,UAAq2B,IAAr2B,UAA02B,IAA12B,UAA+2B,IAA/2B,UAAo3B,IAAp3B,UAAy3B,IAAz3B,UAA83B,IAA93B,UAAm4B,IAAn4B,UAAw4B,IAAx4B,UAA64B,IAA74B,UAAk5B,IAAl5B,UAAu5B,IAAv5B,UAA45B,IAA55B,UAAi6B,IAAj6B,UAAs6B,IAAt6B,UAA26B,IAA36B,UAAg7B,IAAh7B,UAAq7B,IAAr7B,UAA07B,IAA17B,UAA+7B,IAA/7B,UAAo8B,IAAp8B,UAAy8B,IAAz8B,UAA88B,IAA98B,UAAm9B,IAAn9B,UAAw9B,IAAx9B,UAA69B,IAA79B,UAAk+B,IAAl+B,UAAu+B,IAAv+B,UAA4+B,IAA5+B,UAAi/B,IAAj/B,UAAs/B,IAAt/B,UAA2/B,IAA3/B,UAAggC,IAAhgC,UAAqgC,IAArgC,UAA0gC,IAA1gC,UAA+gC,IAA/gC,UAAohC,IAAphC,UAAyhC,IAAzhC,UAA8hC,IAA9hC,UAAmiC,IAAniC,UAAwiC,IAAxiC,UAA6iC,IAA7iC,UAAkjC,IAAljC,UAAujC,IAAvjC,UAA4jC,IAA5jC,UAAikC,IAAjkC,UAAskC,IAAtkC,UAA2kC,IAA3kC,UAAglC,IAAhlC,UAAqlC,IAArlC,UAA0lC,IAA1lC,UAA+lC,IAA/lC,UAAomC,IAApmC,UAAymC,IAAzmC,UAA8mC,IAA9mC,UAAmnC,IAAnnC,UAAwnC,IAAxnC,UAA6nC,IAA7nC,UAAkoC,IAAloC,UAAuoC,IAAvoC,UAA4oC,IAA5oC,UAAipC,IAAjpC,UAAspC,IAAtpC,UAA2pC,IAA3pC,UAAgqC,IAAhqC,UAAqqC,IAArqC,UAA0qC,IAA1qC,UAA+qC,IAA/qC,UAAorC,IAAprC,UAAyrC,IAAzrC,UAA8rC,IAA9rC,UAAmsC,IAAnsC,UAAwsC,IAAxsC,UAA6sC,IAA7sC,UAAktC,IAAltC,UAAutC,IAAvtC,UAA4tC,IAA5tC,UAAiuC,IAAjuC,UAAsuC,IAAtuC,UAA2uC,IAA3uC,UAAgvC,IAAhvC,UAAqvC,IAArvC,UAA0vC,IAA1vC,UAA+vC,IAA/vC,UAAowC,IAApwC,UAAywC,IAAzwC,UAA8wC,IAA9wC,UAAmxC,IAAnxC,UAAwxC,IAAxxC,UAA6xC,IAA7xC,UAAkyC,IAAlyC,UAAuyC,IAAvyC,UAA4yC,IAA5yC,UAAizC,IAAjzC,UAAszC,IAAtzC,UAA2zC,IAA3zC,UAAg0C,IAAh0C,UAAq0C,IAAr0C,UAA00C,IAA10C,UAA+0C,IAA/0C,UAAo1C,IAAp1C,UAAy1C,IAAz1C,UAA81C,IAA91C,UAAm2C,IAAn2C,UAAw2C,IAAx2C,UAA62C,IAA72C,UAAk3C,IAAl3C,UAAu3C,IAAv3C,UAA43C,IAA53C,UAAi4C,IAAj4C,UAAs4C,IAAt4C,UAA24C,IAA34C,UAAg5C,IAAh5C,UAAq5C,IAAr5C,UAA05C,IAA15C,UAA+5C,IAA/5C,UAAo6C,IAAp6C,UAAy6C,IAAz6C,UAA86C,IAA96C,UAAm7C,IAAn7C,UAAw7C,IAAx7C,UAA67C,IAA77C,UAAk8C,IAAl8C,UAAu8C,IAAv8C,UAA48C,IAA58C,UAAi9C,IAAj9C,UAAs9C,IAAt9C,UAA29C,IAA39C,UAAg+C,IAAh+C,UAAq+C,IAAr+C,UAA0+C,IAA1+C,UAA++C,IAA/+C,UAAo/C,IAAp/C,UAAy/C,IAAz/C,UAA8/C,IAA9/C,UAAmgD,IAAngD,UAAwgD,IAAxgD,UAA6gD,IAA7gD,UAAkhD,IAAlhD,UAAuhD,IAAvhD,UAA4hD,IAA5hD,UAAiiD,IAAjiD,UAAsiD,IAAtiD,UAA2iD,IAA3iD,UAAgjD,IAAhjD,UAAqjD,IAArjD,UAA0jD,IAA1jD,UAA+jD,IAA/jD,UAAokD,IAApkD,UAAykD,IAAzkD,UAA8kD,IAA9kD,UAAmlD,IAAnlD,UAAwlD,IAAxlD,UAA6lD,IAA7lD,UAAkmD,IAAlmD,UAAumD,IAAvmD,UAA4mD,IAA5mD,UAAinD,IAAjnD,UAAsnD,IAAtnD,UAA2nD,IAA3nD,UAAgoD,IAAhoD,UAAqoD,IAAroD,UAA0oD,IAA1oD,UAA+oD,IAA/oD,UAAopD,IAAppD,UAAypD,IAAzpD,UAA8pD,IAA9pD,UAAmqD,IAAnqD,UAAwqD,IAAxqD,UAA6qD,IAA7qD,UAAkrD,IAAlrD,UAAurD,IAAvrD,UAA4rD,IAA5rD,UAAisD,IAAjsD,UAAssD,IAAtsD,UAA2sD,IAA3sD,UAAgtD,IAAhtD,UAAqtD,IAArtD,UAA0tD,IAA1tD,UAA+tD,IAA/tD,UAAouD,IAApuD,UAAyuD,IAAzuD,UAA8uD,IAA9uD,UAAmvD,IAAnvD,UAAwvD,IAAxvD,UAA6vD,IAA7vD,UAAkwD,IAAlwD,UAAuwD,IAAvwD,UAA4wD,IAA5wD,UAAixD,IAAjxD,UAAsxD,IAAtxD,UAA2xD,IAA3xD,UAAgyD,IAAhyD,UAAqyD,IAAryD,UAA0yD,IAA1yD,UAA+yD,IAA/yD,UAAozD,IAApzD,UAAyzD,IAAzzD,UAA8zD,IAA9zD,UAAm0D,IAAn0D,UAAw0D,IAAx0D,UAA60D,IAA70D,UAAk1D,IAAl1D,UAAu1D,IAAv1D,UAA41D,IAA51D,UAAi2D,IAAj2D,UAAs2D,IAAt2D,UAA22D,IAA32D,UAAg3D,IAAh3D,UAAq3D,IAAr3D,UAA03D,IAA13D,UAA+3D,IAA/3D,UAAo4D,IAAp4D,UAAy4D,IAAz4D,UAA84D,IAA94D,UAAm5D,IAAn5D,UAAw5D,IAAx5D,UAA65D,IAA75D,UAAk6D,IAAl6D,UAAu6D,IAAv6D,UAA46D,IAA56D,UAAi7D,IAAj7D,UAAs7D,IAAt7D,UAA27D,IAA37D,UAAg8D,IAAh8D,UAAq8D,IAAr8D,UAA08D,IAA18D,UAA+8D,IAA/8D,UAAo9D,IAAp9D,UAAy9D,IAAz9D,UAA89D,IAA99D,UAAm+D,IAAn+D,UAAw+D,IAAx+D,UAA6+D,IAA7+D,UAAk/D,IAAl/D,UAAu/D,IAAv/D,UAA4/D,IAA5/D,UAAigE,IAAjgE,UAAsgE,IAAtgE,UAA2gE,IAA3gE,UAAghE,IAAhhE,UAAqhE,IAArhE,UAA0hE,IAA1hE,UAA+hE,IAA/hE,UAAoiE,IAApiE,UAAyiE,IAAziE,UAA8iE,IAA9iE,UAAmjE,IAAnjE,UAAwjE,IAAxjE,UAA6jE,IAA7jE,UAAkkE,IAAlkE,UAAukE,IAAvkE,UAA4kE,IAA5kE,UAAilE,IAAjlE,UAAslE,IAAtlE,UAA2lE,IAA3lE,UAAgmE,IAAhmE,UAAqmE,IAArmE,UAA0mE,IAA1mE,UAA+mE,IAA/mE,UAAonE,IAApnE,UAAynE,IAAznE,UAA8nE,IAA9nE,UAAmoE,IAAnoE,UAAwoE,IAAxoE,UAA6oE,IAA7oE,UAAkpE,IAAlpE,UAAupE,IAAvpE,UAA4pE,IAA5pE,UAAiqE,IAAjqE,UAAsqE,IAAtqE,UAA2qE,IAA3qE,UAAgrE,IAAhrE,UAAqrE,IAArrE,UAA0rE,IAA1rE,UAA+rE,IAA/rE,UAAosE,IAApsE,UAAysE,IAAzsE,UAA8sE,IAA9sE,UAAmtE,IAAntE,UAAwtE,IAAxtE,UAA6tE,IAA7tE,UAAkuE,IAAluE,UAAuuE,IAAvuE,UAA4uE,IAA5uE,UAAivE,IAAjvE,UAAsvE,IAAtvE,UAA2vE,IAA3vE,UAAgwE,IAAhwE,UAAqwE,IAArwE,UAA0wE,IAA1wE,UAA+wE,IAA/wE,UAAoxE,IAApxE,UAAyxE,IAAzxE,UAA8xE,IAA9xE,UAAmyE,IAAnyE,UAAwyE,IAAxyE,UAA6yE,IAA7yE,UAAkzE,IAAlzE,UAAuzE,IAAvzE,UAA4zE,IAA5zE,UAAi0E,IAAj0E,UAAs0E,IAAt0E,UAA20E,IAA30E,UAAg1E,IAAh1E,UAAq1E,IAAr1E,UAA01E,IAA11E,UAA+1E,IAA/1E,UAAo2E,IAAp2E,UAAy2E,IAAz2E,UAA82E,IAA92E,UAAm3E,IAAn3E,UAAw3E,IAAx3E,UAA63E,IAA73E,UAAk4E,IAAl4E,UAAu4E,IAAv4E,UAA44E,IAA54E,UAAi5E,IAAj5E,UAAs5E,IAAt5E,UAA25E,IAA35E,UAAg6E,IAAh6E,UAAq6E,IAAr6E,UAA06E,IAA16E,UAA+6E,IAA/6E,UAAo7E,IAAp7E,UAAy7E,IAAz7E,UAA87E,IAA97E,UAAm8E,IAAn8E,UAAw8E,IAAx8E,UAA68E,IAA78E,UAAk9E,IAAl9E,UAAu9E,IAAv9E,UAA49E,IAA59E,UAAi+E,IAAj+E,UAAs+E,IAAt+E,UAA2+E,IAA3+E,UAAg/E,IAAh/E,UAAq/E,IAAr/E,UAA0/E,IAA1/E,UAA+/E,IAA//E,UAAohF,IAAphF,UAAyhF,IAAzhF,UAA8hF,IAA9hF,UAAmiF,IAAniF,UAAwiF,IAAxiF,UAA6iF,IAA7iF,UAAkjF,IAAljF,UAAujF,IAAvjF,UAA4jF,IAA5jF,UAAikF,IAAjkF,UAAskF,IAAtkF,UAA2kF,IAA3kF,UAAglF,IAAhlF,UAAqlF,IAArlF,UAA0lF,IAA1lF,UAA+lF,IAA/lF,UAAomF,IAApmF,UAAymF,IAAzmF,UAA8mF,IAA9mF,UAAmnF,IAAnnF,UAAwnF,IAAxnF,UAA6nF,IAA7nF,UAAkoF,IAAloF,UAAuoF,IAAvoF,UAA4oF,IAA5oF,UAAipF,IAAjpF,UAAspF,IAAtpF,UAA2pF,IAA3pF,UAAgqF,IAAhqF,UAAqqF,IAArqF,UAA0qF,IAA1qF,UAA+qF,IAA/qF,UAAorF,IAAprF,UAAyrF,IAAzrF,UAA8rF,IAA9rF,UAAmsF,IAAnsF,UAAwsF,IAAxsF,UAA6sF,IAA7sF,UAAktF,IAAltF,UAAutF,IAAvtF,UAA4tF,IAA5tF,UAAiuF,IAAjuF,UAAsuF,IAAtuF,UAA2uF,IAA3uF,UAAgvF,IAAhvF,UAAqvF,IAArvF,UAA0vF,IAA1vF,UAA+vF,IAA/vF,UAAowF,IAApwF,UAAywF,IAAzwF,UAA8wF,IAA9wF,UAAmxF,IAAnxF,UAAwxF,IAAxxF,UAA6xF,IAA7xF,UAAkyF,IAAlyF,UAAuyF,IAAvyF,UAA4yF,IAA5yF,UAAizF,IAAjzF,UAAszF,IAAtzF,UAA2zF,IAA3zF,UAAg0F,IAAh0F,UAAq0F,IAAr0F,UAA00F,IAA10F,UAA+0F,IAA/0F,UAAo1F,IAAp1F,UAAy1F,IAAz1F,UAA81F,IAA91F,UAAm2F,IAAn2F,UAAw2F,IAAx2F,UAA62F,IAA72F,UAAk3F,IAAl3F,UAAu3F,IAAv3F,UAA43F,IAA53F,UAAi4F,IAAj4F,UAAs4F,IAAt4F,UAA24F,IAA34F,UAAg5F,IAAh5F,UAAq5F,IAAr5F,UAA05F,IAA15F,UAA+5F,IAA/5F,UAAo6F,IAAp6F,UAAy6F,IAAz6F,UAA86F,IAA96F,UAAm7F,IAAn7F,UAAw7F,IAAx7F,UAA67F,IAA77F,UAAk8F,IAAl8F,UAAu8F,IAAv8F,UAA48F,IAA58F,UAAi9F,IAAj9F,UAAs9F,IAAt9F,UAA29F,IAA39F,UAAg+F,IAAh+F,UAAq+F,IAAr+F,UAA0+F,IAA1+F,UAA++F,IAA/+F,UAAo/F,IAAp/F,UAAy/F,IAAz/F,UAA8/F,IAA9/F,UAAmgG,IAAngG,UAAwgG,IAAxgG,UAA6gG,IAA7gG,UAAkhG,IAAlhG,UAAuhG,IAAvhG,UAA4hG,IAA5hG,UAAiiG,IAAjiG,UAAsiG,IAAtiG,UAA2iG,IAA3iG,UAAgjG,IAAhjG,UAAqjG,IAArjG,UAA0jG,IAA1jG,UAA+jG,IAA/jG,UAAokG,IAApkG,UAAykG,IAAzkG,UAA8kG,IAA9kG,UAAmlG,IAAnlG,UAAwlG,IAAxlG,UAA6lG,IAA7lG,UAAkmG,IAAlmG,UAAumG,IAAvmG,UAA4mG,IAA5mG,UAAinG,IAAjnG,UAAsnG,IAAtnG,UAA2nG,IAA3nG,UAAgoG,IAAhoG,UAAqoG,IAAroG,UAA0oG,IAA1oG,UAA+oG,IAA/oG,UAAopG,IAAppG,UAAypG,IAAzpG,UAA8pG,IAA9pG,UAAmqG,IAAnqG,UAAwqG,IAAxqG,UAA6qG,IAA7qG,UAAkrG,IAAlrG,UAAurG,IAAvrG,UAA4rG,IAA5rG,UAAisG,IAAjsG,UAAssG,IAAtsG,UAA2sG,IAA3sG,UAAgtG,IAAhtG,UAAqtG,IAArtG,UAA0tG,IAA1tG,UAA+tG,IAA/tG,UAAouG,IAApuG,UAAyuG,IAAzuG,UAA8uG,IAA9uG,UAAmvG,IAAnvG,UAAwvG,IAAxvG,UAA6vG,IAA7vG,UAAkwG,IAAlwG,UAAuwG,IAAvwG,UAA4wG,IAA5wG,UAAixG,IAAjxG,UAAsxG,IAAtxG,UAA2xG,IAA3xG,UAAgyG,IAAhyG,UAAqyG,IAAryG,UAA0yG,IAA1yG,UAA+yG,IAA/yG,UAAozG,IAApzG,UAAyzG,IAAzzG,UAA8zG,IAA9zG,UAAm0G,IAAn0G,UAAw0G,IAAx0G,UAA60G,IAA70G,UAAk1G,IAAl1G,UAAu1G,IAAv1G,UAA41G,IAA51G,UAAi2G,IAAj2G,UAAs2G,IAAt2G,UAA22G,IAA32G,UAAg3G,IAAh3G,UAAq3G,IAAr3G,UAA03G,IAA13G,UAA+3G,IAA/3G,UAAo4G,IAAp4G,UAAy4G,IAAz4G,UAA84G,IAA94G,UAAm5G,IAAn5G,UAAw5G,IAAx5G,UAA65G,IAA75G,UAAk6G,IAAl6G,UAAu6G,IAAv6G,UAA46G,IAA56G,UAAi7G,IAAj7G,UAAs7G,IAAt7G,UAA27G,IAA37G,UAAg8G,IAAh8G,UAAq8G,IAAr8G,UAA08G,IAA18G,UAA+8G,IAA/8G,UAAo9G,IAAp9G,UAAy9G,IAAz9G,UAA89G,IAA99G,UAAm+G,IAAn+G,UAAw+G,IAAx+G,UAA6+G,IAA7+G,UAAk/G,IAAl/G,UAAu/G,IAAv/G,UAA4/G,IAA5/G,UAAigH,IAAjgH,UAAsgH,IAAtgH,UAA2gH,IAA3gH,UAAghH,IAAhhH,UAAqhH,IAArhH,UAA0hH,IAA1hH,UAA+hH,IAA/hH,UAAoiH,IAApiH,UAAyiH,IAAziH,UAA8iH,IAA9iH,UAAmjH,IAAnjH,UAAwjH,IAAxjH,UAA6jH,IAA7jH,UAAkkH,IAAlkH,UAAukH,IAAvkH,UAA4kH,IAA5kH,UAAilH,IAAjlH,UAAslH,IAAtlH,UAA2lH,IAA3lH,UAAgmH,IAAhmH,UAAqmH,IAArmH,UAA0mH,IAA1mH,UAA+mH,IAA/mH,UAAonH,IAApnH,UAAynH,IAAznH,UAA8nH,IAA9nH,UAAmoH,IAAnoH,UAAwoH,IAAxoH,UAA6oH,IAA7oH,UAAkpH,IAAlpH,UAAupH,IAAvpH,UAA4pH,IAA5pH,UAAiqH,IAAjqH,UAAsqH,IAAtqH,UAA2qH,IAA3qH,UAAgrH,IAAhrH,UAAqrH,IAArrH,UAA0rH,IAA1rH,UAA+rH,IAA/rH,UAAosH,IAApsH,UAAysH,IAAzsH,UAA8sH,IAA9sH,UAAmtH,IAAntH,UAAwtH,IAAxtH,UAA6tH,IAA7tH,UAAkuH,IAAluH,UAAuuH,IAAvuH,UAA4uH,IAA5uH,UAAivH,IAAjvH,UAAsvH,IAAtvH,UAA2vH,IAA3vH,UAAgwH,IAAhwH,UAAqwH,IAArwH,UAA0wH,IAA1wH,UAA+wH,IAA/wH,UAAoxH,IAApxH,UAAyxH,IAAzxH,UAA8xH,IAA9xH,UAAmyH,IAAnyH,UAAwyH,IAAxyH,UAA6yH,IAA7yH,UAAkzH,IAAlzH,UAAuzH,IAAvzH,UAA4zH,IAA5zH,UAAi0H,IAAj0H,UAAs0H,IAAt0H,UAA20H,IAA30H,UAAg1H,IAAh1H,UAAq1H,IAAr1H,UAA01H,IAA11H,UAA+1H,IAA/1H,UAAo2H,IAAp2H,UAAy2H,IAAz2H,UAA82H,IAA92H,UAAm3H,IAAn3H,UAAw3H,IAAx3H,UAA63H,IAA73H,UAAk4H,IAAl4H,UAAu4H,IAAv4H,UAA44H,IAA54H,UAAi5H,IAAj5H,UAAs5H,IAAt5H,UAA25H,IAA35H,UAAg6H,IAAh6H,UAAq6H,IAAr6H,UAA06H,IAA16H,UAA+6H,IAA/6H,UAAo7H,IAAp7H,UAAy7H,IAAz7H,UAA87H,IAA97H,UAAm8H,IAAn8H,UAAw8H,IAAx8H,UAA68H,IAA78H,UAAk9H,IAAl9H,UAAu9H,IAAv9H,UAA49H,IAA59H,UAAi+H,IAAj+H,UAAs+H,IAAt+H,UAA2+H,IAA3+H,UAAg/H,IAAh/H,UAAq/H,IAAr/H,UAA0/H,IAA1/H,UAA+/H,IAA//H,UAAohI,IAAphI,UAAyhI,IAAzhI,UAA8hI,IAA9hI,UAAmiI,IAAniI,UAAwiI,IAAxiI,UAA6iI,IAA7iI,UAAkjI,IAAljI,UAAujI,IAAvjI,UAA4jI,IAA5jI,UAAikI,IAAjkI,UAAskI,IAAtkI,UAA2kI,IAA3kI,UAAglI,IAAhlI,UAAqlI,IAArlI,UAA0lI,IAA1lI,UAA+lI,IAA/lI,UAAomI,IAApmI,UAAymI,IAAzmI,UAA8mI,IAA9mI,UAAmnI,IAAnnI,UAAwnI,IAAxnI,UAA6nI,IAA7nI,UAAkoI,IAAloI,UAAuoI,IAAvoI,UAA4oI,IAA5oI,UAAipI,IAAjpI,UAAspI,IAAtpI,UAA2pI,IAA3pI,UAAgqI,IAAhqI,UAAqqI,IAArqI,UAA0qI,IAA1qI,UAA+qI,IAA/qI,UAAorI,IAAprI,UAAyrI,IAAzrI,UAA8rI,IAA9rI,UAAmsI,IAAnsI,UAAwsI,IAAxsI,UAA6sI,IAA7sI,UAAktI,IAAltI,UAAutI,IAAvtI,UAA4tI,IAA5tI,UAAiuI,IAAjuI,UAAsuI,IAAtuI,UAA2uI,IAA3uI,UAAgvI,IAAhvI,UAAqvI,IAArvI,UAA0vI,IAA1vI,UAA+vI,IAA/vI,UAAowI,IAApwI,UAAywI,IAAzwI,UAA8wI,IAA9wI,UAAmxI,IAAnxI,UAAwxI,IAAxxI,UAA6xI,IAA7xI,UAAkyI,IAAlyI,UAAuyI,IAAvyI,UAA4yI,IAA5yI,UAAizI,IAAjzI,UAAszI,IAAtzI,UAA2zI,IAA3zI,UAAg0I,IAAh0I,UAAq0I,IAAr0I,UAA00I,IAA10I,UAA+0I,IAA/0I,UAAo1I,IAAp1I,UAAy1I,IAAz1I,UAA81I,IAA91I,UAAm2I,IAAn2I,UAAw2I,IAAx2I,UAA62I,IAA72I,UAAk3I,IAAl3I,UAAu3I,IAAv3I,UAA43I,IAA53I,UAAi4I,IAAj4I,UAAs4I,IAAt4I,UAA24I,IAA34I,UAAg5I,IAAh5I,UAAq5I,IAAr5I,UAA05I,IAA15I,UAA+5I,IAA/5I,UAAo6I,IAAp6I,UAAy6I,IAAz6I,UAA86I,IAA96I,UAAm7I,IAAn7I,UAAw7I,IAAx7I,UAA67I,IAA77I,UAAk8I,IAAl8I,UAAu8I,IAAv8I,UAA48I,IAA58I,UAAi9I,IAAj9I,UAAs9I,IAAt9I,UAA29I,IAA39I,UAAg+I,IAAh+I,UAAq+I,IAAr+I,UAA0+I,IAA1+I,UAA++I,IAA/+I,UAAo/I,IAAp/I,UAAy/I,IAAz/I,UAA8/I,IAA9/I,UAAmgJ,IAAngJ,UAAwgJ,IAAxgJ,UAA6gJ,IAA7gJ,UAAkhJ,IAAlhJ,UAAuhJ,IAAvhJ,UAA4hJ,IAA5hJ,UAAiiJ,IAAjiJ,UAAsiJ,IAAtiJ,UAA2iJ,IAA3iJ,UAAgjJ,IAAhjJ,UAAqjJ,IAArjJ,UAA0jJ,IAA1jJ,UAA+jJ,IAA/jJ,UAAokJ,IAApkJ,UAAykJ,IAAzkJ,UAA8kJ,IAA9kJ,UAAmlJ,IAAnlJ,UAAwlJ,IAAxlJ,UAA6lJ,IAA7lJ,UAAkmJ,IAAlmJ,UAAumJ,IAAvmJ,UAA4mJ,IAA5mJ,UAAinJ,IAAjnJ,UAAsnJ,IAAtnJ,UAA2nJ,IAA3nJ,UAAgoJ,IAAhoJ,UAAqoJ,IAAroJ,UAA0oJ,IAA1oJ,UAA+oJ,IAA/oJ,UAAopJ,IAAppJ,UAAypJ,IAAzpJ,UAA8pJ,IAA9pJ,UAAmqJ,IAAnqJ,UAAwqJ,IAAxqJ,UAA6qJ,IAA7qJ,UAAkrJ,IAAlrJ,UAAurJ,IAAvrJ,UAA4rJ,IAA5rJ,UAAisJ,IAAjsJ,UAAssJ,IAAtsJ,UAA2sJ,IAA3sJ,UAAgtJ,IAAhtJ,UAAqtJ,IAArtJ,UAA0tJ,IAA1tJ,UAA+tJ,IAA/tJ,UAAouJ,IAApuJ,UAAyuJ,IAAzuJ,UAA8uJ,IAA9uJ,UAAmvJ,IAAnvJ,UAAwvJ,IAAxvJ,UAA6vJ,IAA7vJ,UAAkwJ,IAAlwJ,UAAuwJ,IAAvwJ,UAA4wJ,IAA5wJ,UAAixJ,IAAjxJ,UAAsxJ,IAAtxJ,UAA2xJ,IAA3xJ,UAAgyJ,IAAhyJ,UAAqyJ,IAAryJ,UAA0yJ,IAA1yJ,UAA+yJ,IAA/yJ,UAAozJ,IAApzJ,UAAyzJ,IAAzzJ,UAA8zJ,IAA9zJ,UAAm0J,IAAn0J,UAAw0J,IAAx0J,UAA60J,IAA70J,UAAk1J,IAAl1J,UAAu1J,IAAv1J,UAA41J,IAA51J,UAAi2J,IAAj2J,UAAs2J,IAAt2J,UAA22J,IAA32J,UAAg3J,IAAh3J,UAAq3J,IAAr3J,UAA03J,IAA13J,UAA+3J,IAA/3J,UAAo4J,IAAp4J,UAAy4J,IAAz4J,UAA84J,IAA94J,UAAm5J,IAAn5J,UAAw5J,IAAx5J,UAA65J,IAA75J,UAAk6J,IAAl6J,UAAu6J,IAAv6J,UAA46J,IAA56J,UAAi7J,IAAj7J,UAAs7J,IAAt7J,UAA27J,IAA37J,UAAg8J,IAAh8J,UAAq8J,IAAr8J,UAA08J,IAA18J,UAA+8J,IAA/8J,UAAo9J,IAAp9J,UAAy9J,IAAz9J,UAA89J,IAA99J,UAAm+J,IAAn+J,UAAw+J,IAAx+J,UAA6+J,IAA7+J,UAAk/J,IAAl/J,UAAu/J,IAAv/J,UAA4/J,IAA5/J,UAAigK,IAAjgK,UAAsgK,IAAtgK,UAA2gK,IAA3gK,UAAghK,IAAhhK,UAAqhK,IAArhK,UAA0hK,IAA1hK,UAA+hK,IAA/hK,UAAoiK,IAApiK,UAAyiK,IAAziK,UAA8iK,IAA9iK,UAAmjK,IAAnjK,UAAwjK,IAAxjK,UAA6jK,IAA7jK,UAAkkK,IAAlkK,UAAukK,IAAvkK,UAA4kK,IAA5kK,UAAilK,IAAjlK,UAAslK,IAAtlK,UAA2lK,IAA3lK,UAAgmK,IAAhmK,UAAqmK,IAArmK,UAA0mK,IAA1mK,UAA+mK,IAA/mK,UAAonK,IAApnK,UAAynK,IAAznK,UAA8nK,IAA9nK,UAAmoK,IAAnoK,UAAwoK,IAAxoK,UAA6oK,IAA7oK,UAAkpK,IAAlpK,UAAupK,IAAvpK,UAA4pK,IAA5pK,UAAiqK,IAAjqK,UAAsqK,IAAtqK,UAA2qK,IAA3qK,UAAgrK,IAAhrK,UAAqrK,IAArrK,UAA0rK,IAA1rK,UAA+rK,IAA/rK,UAAosK,IAApsK,UAAysK,IAAzsK,UAA8sK,IAA9sK,UAAmtK,IAAntK,UAAwtK,IAAxtK,UAA6tK,IAA7tK,UAAkuK,IAAluK,UAAuuK,IAAvuK,UAA4uK,IAA5uK,UAAivK,IAAjvK,UAAsvK,IAAtvK,UAA2vK,IAA3vK,UAAgwK,IAAhwK,UAAqwK,IAArwK,UAA0wK,IAA1wK,UAA+wK,IAA/wK,UAAoxK,IAApxK,UAAyxK,IAAzxK,UAA8xK,IAA9xK,UAAmyK,IAAnyK,UAAwyK,IAAxyK,UAA6yK,IAA7yK,UAAkzK,IAAlzK,UAAuzK,IAAvzK,UAA4zK,IAA5zK,UAAi0K,IAAj0K,UAAs0K,IAAt0K,UAA20K,IAA30K,UAAg1K,IAAh1K,UAAq1K,IAAr1K,UAA01K,IAA11K,UAA+1K,IAA/1K,UAAo2K,IAAp2K,UAAy2K,IAAz2K,UAA82K,IAA92K,UAAm3K,IAAn3K,UAAw3K,IAAx3K,UAA63K,IAA73K,UAAk4K,IAAl4K,UAAu4K,IAAv4K,UAA44K,IAA54K,UAAi5K,IAAj5K,UAAs5K,IAAt5K,UAA25K,IAA35K,UAAg6K,IAAh6

W,gB;QACX,wBAAa,SAAb,gB;UAAa,WAAb,UAAa,SAAb,O;UACI,IAAI,CAAC,UAAU,iBAAV,CAAL,C;YACI,K;UACJ,IAAK,WAAI,iBAAJ,C;;QAET,OAAO,I;O;KAZX,C;IAeA,4B;MAII,eAAe,CAAC,mBAAO,CAAP,AAD,IAAa,CAAb,I;MACf,IAAI,WAAW,CAAf,C;QAAkB,M;MAClB,mBAAmB,wB;MACnB,iBAAc,CAAd,WAAiB,QAAjB,U;QACI,UAAU,UAAK,KAAL,C;QACV,UAAK,KAAL,IAAc,C;QACd,UAAK,KAAL,IAAc,UAAK,YAAL,C;QACd,UAAK,YAAL,IAAqB,G;QACrB,mC;;K;IAIR,8B;MAII,eAAe,CAAC,mBAAO,CAAP,AAD,IAAa,CAAb,I;MACf,IAAI,WAAW,CAAf,C;QAAkB,M;MAClB,mBAAmB,0B;MACnB,iBAAc,CAAd,WAAiB,QAAjB,U;QACI,UAAU,UAAK,KAAL,C;QACV

AAd,UAA8B,QAA9B,U;QACI,UAAU,UAAK,KAAL,C;QACV,UAAK,KAAL,IAAc,UAAK,YAAL,C;QACd,UAAK,YAAL,IAAqB,G;QACrB,mC;;K;IAIR,mD;MAWI,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;MACb,eAAe,CAAC,YAAY,OAAZ,IAAD,IAAwB,CAAxB,I;MACf,IAAI,cAAa,QAAjB,B,C;QAA2B,M;MAC3B,mBAAmB,UAAU,CAAV,I;MACnB,iBAAc,SAAd,UAA8B,QAA9B,U;QACI,UAAU,UAAK,KAAL,C;QACV,UAAK,KAAL,IAAc,UAAK,YAAL,C;QACd,UAAK,YAAL,IAAqB,G;QACrB,mC;;K;IAIR,mD;MAWI,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;MACb,eAAe,CAAC,YAAY,OAAZ,IAAD,IAAwB,CAAxB,I;MACf,IAAI,cAAa,QAAjB,B,C;QAA2B,M;MAC3B,mBAAmB,UAAU,CAAV,I;MACnB,iBAAc,SAAd,UAA8B,QAA9B,U;QACI,UAAU,UAAK,KAAL,C;QACV,UAAK,KAAL,IAAc,UAAK,YAAL,C;QACd,UAAK,YAAL,IAAqB,G;QACrB,mC;;K;IAIR,6B;MAWI,IA+nEO,qBAAqB,CAA/B,Ef,C;QAAiC,eAAO,O;MACxC,W;MAAA,WAAK,IAAL,C;QAAK,IAAL,IAAK,C;QAAL,IAAK,IAAL,IAAK,C;QAAL,IAAK,IAAL,IAAK,C;QAAL,I

U,I;;K;IAIlB,uC;MAOI,aAAU,0BAAV,OAA2B,CAA3B,M;QACI,QAAQ,MAAO,iBAAQ,IAAI,CAAJ,IAAR,C;QACf,WAAW,UAAK,CAAL,C;QACX,UAAK,CAAL,IAAU,UAAK,CAAL,C;QACV,UAAK,CAAL,IAAU

;W;S;OA4DI,C;MhBqrMf,sC;QAQI,OAAO,sBgB7rMP,eAAW,iBhB6rMiB,QgB7rMjB,CAAX,ChB6rMO,C;O;KARX,C;wFAWA,yB;MAAA,wD;MgBvsMA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CAA3B,C;W;S;OA4DI,C;MhBgsMf,sC;QAMI,OAAO,sBgBtsMP,eAAW,iBhBssMiB,QgBtsMjB,CAAX,ChBssMO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBhtMA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBysMf,sC;QAMI,OAAO,sBgB/sMP,eAAW,iBh+sMiB,Qg/sMjB,CAAX,Ch+sMO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBztMA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBktMf,sC;QAMI,OAAO,sBgBxtMP,eAAW,iBhBwtMiB,QgBxtMjB,CAAX,ChBwtMO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBluMA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhB2tMf,sC;QAMI,OAAO,sBgBjuMP,eAAW,iBhBiuMiB,QgBjuMjB,CAAX,ChBiuMO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgB3uMA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBouMf,sC;QAMI,OAAO,sBgB1uMP,eAAW,iBhB0uMiB,QgB1uMjB,CAAX,ChB0uMO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBpvMA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhB6vMf,sC;QAMI,OAAO,sBgBnwMP,eAAW,iBhBmwMiB,QgBnwMjB,CAAX,ChBmwMO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgB7wMA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBsxMf,sC;QAMI,OAAO,sBgB5xMP,eAAW,iBhB4xMiB,QgB5xMjB,CAAX,ChB4xMO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBtyMA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhB+yMf,sC;QAMI,OAAO,sBgBrzMP,eAAW,iBhBqzMiB,QgBrzMjB,CAAX,ChBqzMO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgB/zMA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBw0Mf,sC;QAMI,OAAO,sBgB90MP,eAAW,iBhB80MiB,QgB90MjB,CAAX,ChB80MO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBx1MA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBi2Mf,sC;QAMI,OAAO,sBgBv2MP,eAAW,iBhBu2MiB,QgBv2MjB,CAAX,ChBu2MO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBj3MA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhB04Mf,sC;QAMI,OAAO,sBgBh5MP,eAAW,iBhBg5MiB,QgBh5MjB,CAAX,ChBg5MO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgB15MA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBm6Mf,sC;QAMI,OAAO,sBgBz6MP,eAAW,iBhBy6MiB,QgBz6MjB,CAAX,ChBy6MO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBn7MA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhB48Mf,sC;QAMI,OAAO,sBgBl9MP,eAAW,iBhBk9MiB,QgBl9MjB,CAAX,ChBk9MO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgB59MA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBq+Mf,sC;QAMI,OAAO,sBgB3+MP,eAAW,iBhB2+MiB,QgB3+MjB,CAAX,ChB2+MO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBr/MA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhB8/Mf,sC;QAMI,OAAO,sBgBpgNP,eAAW,iBhBogNiB,QgBpgNjB,CAAX,ChBogNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgB9gNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBuhNf,sC;QAMI,OAAO,sBgB7hNP,eAAW,iBhB6hNiB,QgB7hNjB,CAAX,ChB6hNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBviNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBgjNf,sC;QAMI,OAAO,sBgBtjNP,eAAW,iBhBsjNiB,QgBtjNjB,CAAX,ChBsjNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBhkNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBykNf,sC;QAMI,OAAO,sBgB/kNP,eAAW,iBhB+kNiB,QgB/kNjB,CAAX,ChB+kNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBzlNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBsmNf,sC;QAMI,OAAO,sBgB5mNP,eAAW,iBhB4mNiB,QgB5mNjB,CAAX,ChB4mNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBtnNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBmoNf,sC;QAMI,OAAO,sBgBzoNP,eAAW,iBhByoNiB,QgBzoNjB,CAAX,ChByoNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBnpNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBgqNf,sC;QAMI,OAAO,sBgBtqNP,eAAW,iBhBsqNiB,QgBtqNjB,CAAX,ChBsqNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBhrNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBysNf,sC;QAMI,OAAO,sBgB/sNP,eAAW,iBhB+sNiB,QgB/sNjB,CAAX,ChB+sNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBztNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBsuNf,sC;QAMI,OAAO,sBgB5uNP,eAAW,iBhB4uNiB,QgB5uNjB,CAAX,ChB4uNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBtvNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBmwNf,sC;QAMI,OAAO,sBgBzwNP,eAAW,iBhBywNiB,QgBzwNjB,CAAX,ChBywNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBnxNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBgyNf,sC;QAMI,OAAO,sBgBtyNP,eAAW,iBhBsyNiB,QgBtyNjB,CAAX,ChBsyNO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBhzNA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBy0Nf,sC;QAMI,OAAO,sBgB/0NP,eAAW,iBhB+0NiB,QgB/0NjB,CAAX,ChB+0NO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBz1NA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBs2Nf,sC;QAMI,OAAO,sBgB52NP,eAAW,iBhB42NiB,QgB52NjB,CAAX,ChB42NO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBt3NA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBm4Nf,sC;QAMI,OAAO,sBgBz4NP,eAAW,iBhBy4NiB,QgBz4NjB,CAAX,ChBy4NO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBn5NA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBg6Nf,sC;QAMI,OAAO,sBgBt6NP,eAAW,iBhBs6NiB,QgBt6NjB,CAAX,ChBs6NO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBh7NA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBy8Nf,sC;QAMI,OAAO,sBgB/8NP,eAAW,iBhB+8NiB,QgB/8NjB,CAAX,ChB+8NO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBz9NA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBs+Nf,sC;QAMI,OAAO,sBgB5+NP,eAAW,iBhB4+NiB,QgB5+NjB,CAAX,ChB4+NO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBtgOA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBmhOf,sC;QAMI,OAAO,sBgBzhOP,eAAW,iBhByhOiB,QgBzhOjB,CAAX,ChByhOO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBnjOA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBgkOf,sC;QAMI,OAAO,sBgBtkOP,eAAW,iBhBskOiB,QgBtkOjB,CAAX,ChBskOO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBhlOA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBymOf,sC;QAMI,OAAO,sBgB/mOP,eAAW,iBhB+mOiB,QgB/mOjB,CAAX,ChB+mOO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBzoOA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBspOf,sC;QAMI,OAAO,sBgB5pOP,eAAW,iBhB4pOiB,QgB5pOjB,CAAX,ChB4pOO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBtqOA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBmrOf,sC;QAMI,OAAO,sBgBzrOP,eAAW,iBhByrOiB,QgBzrOjB,CAAX,ChByrOO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBnsOA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBgtOf,sC;QAMI,OAAO,sBgBttOP,eAAW,iBhBstOiB,QgBttOjB,CAAX,ChBstOO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBhuOA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBywOf,sC;QAMI,OAAO,sBgB/wOP,eAAW,iBhB+wOiB,QgB/wOjB,CAAX,ChB+wOO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBzxOA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBszOf,sC;QAMI,OAAO,sBgB5zOP,eAAW,iBhB4zOiB,QgB5zOjB,CAAX,ChB4zOO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBt0OA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBm1Of,sC;QAMI,OAAO,sBgBz1OP,eAAW,iBhBy1OiB,QgBz1OjB,CAAX,ChBy1OO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBn2OA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBg3Of,sC;QAMI,OAAO,sBgBt3OP,eAAW,iBhBs3OiB,QgBt3OjB,CAAX,ChBs3OO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBh4OA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBy4Of,sC;QAMI,OAAO,sBgB/4OP,eAAW,iBhB+4OiB,QgB/4OjB,CAAX,ChB+4OO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBz5OA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBs6Of,sC;QAMI,OAAO,sBgB56OP,eAAW,iBhB46OiB,QgB56OjB,CAAX,ChB46OO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBt7OA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBm8Of,sC;QAMI,OAAO,sBgBz8OP,eAAW,iBhBy8OiB,QgBz8OjB,CAAX,ChBy8OO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBn9OA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBg+Of,sC;QAMI,OAAO,sBgBt+OP,eAAW,iBhBs+OiB,QgBt+OjB,CAAX,ChBs+OO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBh/OA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBy/Of,sC;QAMI,OAAO,sBgB//OP,eAAW,iBhB+/OiB,QgB//OjB,CAAX,ChB+/OO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBzgPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBshPf,sC;QAMI,OAAO,sBgB5hPP,eAAW,iBhB4hPiB,QgB5hPjB,CAAX,ChB4hPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBtiPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBmjPf,sC;QAMI,OAAO,sBgBzjPP,eAAW,iBhByjPiB,QgBzjPjB,CAAX,ChByjPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBnkPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBglPf,sC;QAMI,OAAO,sBgBtlPP,eAAW,iBhBslPiB,QgBtlPjB,CAAX,ChBslPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBhmPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBymPf,sC;QAMI,OAAO,sBgB/mPP,eAAW,iBhB+mPiB,QgB/mPjB,CAAX,ChB+mPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBznPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBsoPf,sC;QAMI,OAAO,sBgB5oPP,eAAW,iBhB4oPiB,QgB5oPjB,CAAX,ChB4oPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBtpPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBmqPf,sC;QAMI,OAAO,sBgBzqPP,eAAW,iBhByqPiB,QgBzqPjB,CAAX,ChByqPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBnrPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBgrPf,sC;QAMI,OAAO,sBgBtrPP,eAAW,iBhBsrPiB,QgBtrPjB,CAAX,ChBsrPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBhsPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBysPf,sC;QAMI,OAAO,sBgB/sPP,eAAW,iBhB+sPiB,QgB/sPjB,CAAX,ChB+sPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBztPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBsuPf,sC;QAMI,OAAO,sBgB5uPP,eAAW,iBhB4uPiB,QgB5uPjB,CAAX,ChB4uPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBtvPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBmwPf,sC;QAMI,OAAO,sBgBzwPP,eAAW,iBhBywPiB,QgBzwPjB,CAAX,ChBywPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBnxPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBgxPf,sC;QAMI,OAAO,sBgBtxPP,eAAW,iBhBsxPiB,QgBtxPjB,CAAX,ChBsxPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBhyPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhByyPf,sC;QAMI,OAAO,sBgB/yPP,eAAW,iBhB+yPiB,QgB/yPjB,CAAX,ChB+yPO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBzzPA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBs0Pf,sC;QAMI,OAAO,sBgB50PP,eAAW,iBhB40PiB,QgB50PjB,CAAX,ChB40PO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBt1PA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBm2Pf,sC;QAMI,OAAO,sBgBz2PP,eAAW,iBhBy2PiB,QgBz2PjB,CAAX,ChBy2PO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBn3PA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBg3Pf,sC;QAMI,OAAO,sBgBt3PP,eAAW,iBhBs3PiB,QgBt3PjB,CAAX,ChBs3PO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBh4PA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBy4Pf,sC;QAMI,OAAO,sBgB/4PP,eAAW,iBhB+4PiB,QgB/4PjB,CAAX,ChB+4PO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBz5PA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBs6Pf,sC;QAMI,OAAO,sBgB56PP,eAAW,iBhB46PiB,QgB56PjB,CAAX,ChB46PO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBt7PA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBm8Pf,sC;QAMI,OAAO,sBgBz8PP,eAAW,iBhBy8PiB,QgBz8PjB,CAAX,ChBy8PO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBn9PA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBg9Pf,sC;QAMI,OAAO,sBgBt9PP,eAAW,iBhBs9PiB,QgBt9PjB,CAAX,ChBs9PO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBh+PA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBy+Pf,sC;QAMI,OAAO,sBgB/+PP,eAAW,iBhB++PiB,QgB/+PjB,CAAX,ChB++PO,C;O;KANX,C;wFASA,yB;MAAA,wD;MgBzgQA,sC;MAAA,oC;MAAA,uBAOe,yB;QArEf,8D;eAqEe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EA2B,SA4DM,CA5DN,CA3B,C;W;S;OA4DI,C;MhBshQf,sC;QAMI,OAAO,sBgB5hQP,eAAW,iBhB4hQiB,QgB5hQjB,CAAX,ChB4hQO,C;O;KANX,C;wFASA

AA,oC;MAAA,iCAOe,yB;QAxFf,8D;eAwFe,4B;UAAA,uB;YAAU,eAAsB,gB;YAAtB,OA/Ed,cAAc,SA+EgB,CA/EhB,CAAd,EAA2B,SA+EM,CA/EN,CAA3B,C;W;S;OA+EI,C;MhB+yMf,sC;QAII,OAAO,sBgBnzMP,eAAW,2BhBmzM2B,QgBnzM3B,CAAX,ChBmzMO,C;O;KAJX,C;IAOA,qC;MAMI,OAAO,sBAAW,cAAX,C;K;IAGX,uC;MAIoB,kBel1KQ,iB;Mfk1KA,iB;MAAxB,OAAiC,WiBx2M1B,WjBw2M0B,C;K;IAGrC,uC;MAIoB,kBe/0KQ,iB;Mf+0KA,iB;MAAxB,OAAiC,WiB/2M1B,WjB+2M0B,C;K;IAGrC,uC;MAIoB,kBe50KQ,iB;Mf40KA,iB;MAAxB,OAAiC,WiBt3M1B,WjBs3M0B,C;K;IAGrC,uC;MAIoB,kBAAT,oB;MAAiB,iB;MAAxB,OAAiC,WiBl5M1B,WjBk5M0B,C;K;IAGrC,2C;MAMI,OAAO,sBAAW,cAAX,C;K;IAGvC,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBl6M9C,WjBk6M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBz6M9C,WjBy6M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBh7M9C,WjBg7M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBv7M9C,WjBu7M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB97M9C,WjB87M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBr8M9C,WjBq8M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB58M9C,WjB48M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBn9M9C,WjBm9M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB19M9C,WjB09M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBj+M9C,WjBi+M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBx+M9C,WjBw+M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/+M9C,WjB++M8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBtgN9C,WjBsgN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB7gN9C,WjB6gN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBphN9C,WjBohN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB3hN9C,WjB2hN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBliN9C,WjBkiN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBziN9C,WjByiN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBhjN9C,WjBgjN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBvjN9C,WjBujN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB9jN9C,WjB8jN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBrkN9C,WjBqkN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB5kN9C,WjB4kN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBnlN9C,WjBmlN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB1lN9C,WjB0lN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBjmN9C,WjBimN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBxmN9C,WjBwmN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/mN9C,WjB+mN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBtnN9C,WjBsnN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB7nN9C,WjB6nN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBpoN9C,WjBooN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB3oN9C,WjB2oN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBlpN9C,WjBkpN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBzpN9C,WjBypN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBhqN9C,WjBgqN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBvqN9C,WjBuqN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB9qN9C,WjB8qN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBrrN9C,WjBqrN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB5rN9C,WjB4rN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBnsN9C,WjBmsN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB1sN9C,WjB0sN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBjtN9C,WjBitN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBxtN9C,WjBwtN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/tN9C,WjB+tN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBtuN9C,WjBsuN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB7uN9C,WjB6uN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBpvN9C,WjBovN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB3vN9C,WjB2vN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBlwN9C,WjBkwN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBzwN9C,WjBywN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBhxN9C,WjBgxN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBvxN9C,WjBuxN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB9xN9C,WjB8xN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBrxN9C,WjBqxN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB5xN9C,WjB4xN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBnyN9C,WjBmyN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB1yN9C,WjB0yN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBjzN9C,WjBizN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBxzN9C,WjBwzN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/zN9C,WjB+zN8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBt0N9C,WjBs0N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB70N9C,WjB60N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBp1N9C,WjBo1N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB31N9C,WjB21N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBl2N9C,WjBk2N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBz2N9C,WjBy2N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBh3N9C,WjBg3N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBv3N9C,WjBu3N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB93N9C,WjB83N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBr4N9C,WjBq4N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB54N9C,WjB44N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBn5N9C,WjBm5N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB15N9C,WjB05N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBj6N9C,WjBi6N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBx6N9C,WjBw6N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/6N9C,WjB+6N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBt7N9C,WjBs7N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB77N9C,WjB67N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBp8N9C,WjBo8N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB38N9C,WjB28N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBl9N9C,WjBk9N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBz9N9C,WjBy9N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBh+N9C,WjBg+N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBv+N9C,WjBu+N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB9+N9C,WjB8+N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBr/N9C,WjBq/N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB5/N9C,WjB4/N8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBngO9C,WjBmgO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB1gO9C,WjB0gO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBjhO9C,WjBihO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBxhO9C,WjBwhO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/hO9C,WjB+hO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBtiO9C,WjBsiO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB7iO9C,WjB6iO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBpjO9C,WjBojO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB3jO9C,WjB2jO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBlkO9C,WjBkkO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBzkO9C,WjBykO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBhlO9C,WjBglO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBvlO9C,WjBulO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB9lO9C,WjB8lO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBrmO9C,WjBqmO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB5mO9C,WjB4mO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBnnO9C,WjBmnO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB1nO9C,WjB0nO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBjoO9C,WjBioO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBxoO9C,WjBwoO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/oO9C,WjB+oO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBtpO9C,WjBspO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB7pO9C,WjB6pO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBpqO9C,WjBoqO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB3qO9C,WjB2qO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBlrO9C,WjBkrO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBzrO9C,WjByrO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBhsO9C,WjBgsO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBvsO9C,WjBusO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB9sO9C,WjB8sO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBrtO9C,WjBqtO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB5tO9C,WjB4tO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBnuO9C,WjBmuO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB1uO9C,WjB0uO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBjvO9C,WjBivO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBxvO9C,WjBwvO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/vO9C,WjB+vO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBtwO9C,WjBswO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB7wO9C,WjB6wO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBpxO9C,WjBoxO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB3xO9C,WjB2xO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBlyO9C,WjBkyO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBzyO9C,WjByyO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBhzO9C,WjBgzO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBvzO9C,WjBuzO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB9zO9C,WjB8zO8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBr0O9C,WjBq0O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB50O9C,WjB40O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBn1O9C,WjBm1O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB11O9C,WjB01O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBj2O9C,WjBi2O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBx2O9C,WjBw2O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/2O9C,WjB+2O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBt3O9C,WjBs3O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB73O9C,WjB63O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBp4O9C,WjBo4O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB34O9C,WjB24O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBl5O9C,WjBk5O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBz5O9C,WjBy5O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBh6O9C,WjBg6O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBv6O9C,WjBu6O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB96O9C,WjB86O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBr7O9C,WjBq7O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB57O9C,WjB47O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBn8O9C,WjBm8O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB18O9C,WjB08O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBj9O9C,WjBi9O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBx9O9C,WjBw9O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/9O9C,WjB+9O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBt+O9C,WjBs+O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB7+O9C,WjB6+O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBp/O9C,WjBo/O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB3/O9C,WjB2/O8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBlgP9C,WjBkgP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBzgP9C,WjBygP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBhhP9C,WjBghP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBvhP9C,WjBuhP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB9hP9C,WjB8hP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBriP9C,WjBqiP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB5iP9C,WjB4iP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBnjP9C,WjBmjP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB1jP9C,WjB0jP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBjkP9C,WjBikP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBxkP9C,WjBwkP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/kP9C,WjB+kP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBtlP9C,WjBslP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB7lP9C,WjB6lP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBpmP9C,WjBomP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB3mP9C,WjB2mP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBlnP9C,WjBknP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBznP9C,WjBynP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBhoP9C,WjBgoP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBvoP9C,WjBuoP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB9oP9C,WjB8oP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBrpP9C,WjBqpP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB5pP9C,WjB4pP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBnqP9C,WjBmqP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB1qP9C,WjB0qP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBjrP9C,WjBirP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBxrP9C,WjBwrP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/rP9C,WjB+rP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBtsP9C,WjBssP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB7sP9C,WjB6sP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBptP9C,WjBotP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB3tP9C,WjB2tP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBluP9C,WjBkuP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBzuP9C,WjByuP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBhvP9C,WjBgvP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBvvP9C,WjBuvP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB9vP9C,WjB8vP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBrwP9C,WjBqwP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB5wP9C,WjB4wP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBnxP9C,WjBmxP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB1xP9C,WjB0xP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBjyP9C,WjBiyP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBxyP9C,WjBwyP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/yP9C,WjB+yP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBtzP9C,WjBszP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB7zP9C,WjB6zP8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBp0P9C,WjBo0P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB30P9C,WjB20P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBl1P9C,WjBk1P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBz1P9C,WjBy1P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBh2P9C,WjBg2P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBv2P9C,WjBu2P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB92P9C,WjB82P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBr3P9C,WjBq3P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB53P9C,WjB43P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBn4P9C,WjBm4P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB14P9C,WjB04P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBj5P9C,WjBi5P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBx5P9C,WjBw5P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB/5P9C,WjB+5P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBt6P9C,WjBs6P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB76P9C,WjB66P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBp7P9C,WjBo7P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB37P9C,WjB27P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBl8P9C,WjBk8P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBz8P9C,WjBy8P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBh9P9C,WjBg9P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBv9P9C,WjBu9P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB99P9C,WjB89P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiBr+P9C,WjBq+P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD,OiB5+P9C,WjB4+P8C,C;K;IAGzD,6C;MAI0B,kBAAf,yB;MAAuB,iC;MAA9B,OAAqD

U,OAAV,C;UOx+QnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QP8zPA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOv/QnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QP60PA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B/,CAAU,OAAV,C;UOtgRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QP41PA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B/,CAAU,OAAV,C;UOrhRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QP22PA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOpiRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QP03PA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOnjRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPg5PA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOklRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPs7PA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOjlRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPo8PA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MA4qBA,oC;MAAA,gC;MA5qBA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOhmRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPs9PA,OA4qBO,W;O;KAxrBX,C;4FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOmoRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPy+PA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOqqRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QP4gQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOstRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QP+iQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOuvRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPkmQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOxxRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPqpQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO0zRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO2zRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO4zRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO6zRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO8zRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO+zRnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOg0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOi0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOk0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOm0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOo0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOq0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOs0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOu0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOw0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOy0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO00RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO20RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO40RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO60RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO80RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO+0RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOg1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOi1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOk1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOm1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOo1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOq1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOs1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOu1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOw1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOy1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO01RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO21RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO41RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO61RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO81RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO+1RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOg2RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOi2RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOk2RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOm2RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOo2RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOq2RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOs2RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOu2RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOw2RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UOy2RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO02RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAY,mBAAoB,QAApB,C;QAyqBH,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA1qB8C,SA0qB/B,B,CAAU,OAAV,C;UO22RnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QPwsQA,OA4qBO,W;O;KAxrBX,C;0FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1

;MAAA,gC;MA7QA,yC;QAWI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAuB,QAAAvB,C;QA2QL,Q;QAAhB,iD;UAAgB,cAAhB,0B;UACI,WAAY,aA5QuC,WA4QnC,CAAY,oBAAZ,CAAJ,EAA0B,oBAA1B,C;;QA5QhB,OA8QO,W;O;KA1RX,C;8FAeA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA6QL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA9QoC,WA8QhC,CAAY,OAAZ,CAAJ,EA9QiD,cA8QvB,CAAe,OAAf,CAA1B,C;;QA9QhB,OAgRO,W;O;KA3RX,C;8FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA+QL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAhRoC,WAgRhC,CAAY,OAAZ,CAAJ,EAhRiD,cAgRvB,CAAe,OAAf,CAA1B,C;;QAhRhB,OAkRO,W;O;KA7RX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAiRL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAlRoC,WAkRhC,CAAY,OAAZ,CAAJ,EAlRiD,cAkRvB,CAAe,OAAf,CAA1B,C;;QAlRhB,OAoRO,W;O;KA/RX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAmRL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aApRoC,WAoRhC,CAAY,OAAZ,CAAJ,EApRiD,cAoRvB,CAAe,OAAf,CAA1B,C;;QApRhB,OAsRO,W;O;KAjSX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAqRL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAtRoC,WAsRhC,CAAY,OAAZ,CAAJ,EAtRiD,cAsRvB,CAAe,OAAf,CAA1B,C;;QAtRhB,OAwRO,W;O;KAnSX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAuRL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAxRoC,WAwRhC,CAAY,OAAZ,CAAJ,EAxRiD,cAwRvB,CAAe,OAAf,CAA1B,C;;QAxRhB,OA0RO,W;O;KArSX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAyRL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA1RoC,WA0RhC,CAAY,OAAZ,CAAJ,EA1RiD,cA0RvB,CAAe,OAAf,CAA1B,C;;QA1RhB,OA4RO,W;O;KAvSX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA2RL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA5RoC,WA4RhC,CAAY,OAAZ,CAAJ,EA5RiD,cA4RvB,CAAe,OAAf,CAA1B,C;;QA5RhB,OA8RO,W;O;KAzSX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA6RL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA9RoC,WA8RhC,CAAY,OAAZ,CAAJ,EA9RiD,cA8RvB,CAAe,OAAf,CAA1B,C;;QA9RhB,OAgSO,W;O;KA3SX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA+RL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAhSoC,WAgShC,CAAY,OAAZ,CAAJ,EAhSiD,cAgSvB,CAAe,OAAf,CAA1B,C;;QAhShB,OAkSO,W;O;KA7SX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAiSL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAlSoC,WAkShC,CAAY,OAAZ,CAAJ,EAlSiD,cAkSvB,CAAe,OAAf,CAA1B,C;;QAlShB,OAoSO,W;O;KA/SX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAmSL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aApSoC,WAoShC,CAAY,OAAZ,CAAJ,EApSiD,cAoSvB,CAAe,OAAf,CAA1B,C;;QApShB,OAsSO,W;O;KAjTX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAqSL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAtSoC,WAsShC,CAAY,OAAZ,CAAJ,EAtSiD,cAsSvB,CAAe,OAAf,CAA1B,C;;QAtShB,OAwSO,W;O;KAnTX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAuSL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAxSoC,WAwShC,CAAY,OAAZ,CAAJ,EAxSiD,cAwSvB,CAAe,OAAf,CAA1B,C;;QAxShB,OA0SO,W;O;KArTX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAySL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA1SoC,WA0ShC,CAAY,OAAZ,CAAJ,EA1SiD,cA0SvB,CAAe,OAAf,CAA1B,C;;QA1ShB,OA4SO,W;O;KAvTX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA2SL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA5SoC,WA4ShC,CAAY,OAAZ,CAAJ,EA5SiD,cA4SvB,CAAe,OAAf,CAA1B,C;;QA5ShB,OA8SO,W;O;KAzTX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA6SL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA9SoC,WA8ShC,CAAY,OAAZ,CAAJ,EA9SiD,cA8SvB,CAAe,OAAf,CAA1B,C;;QA9ShB,OAgTO,W;O;KA3TX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA+SL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAhToC,WAgThC,CAAY,OAAZ,CAAJ,EAhTiD,cAgTvB,CAAe,OAAf,CAA1B,C;;QAhThB,OAkTO,W;O;KA7TX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAiTL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAlToC,WAkThC,CAAY,OAAZ,CAAJ,EAlTiD,cAkTvB,CAAe,OAAf,CAA1B,C;;QAlThB,OAoTO,W;O;KA/TX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAmTL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aApToC,WAoThC,CAAY,OAAZ,CAAJ,EApTiD,cAoTvB,CAAe,OAAf,CAA1B,C;;QApThB,OAsTO,W;O;KAjUX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAqTL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAtToC,WAsThC,CAAY,OAAZ,CAAJ,EAtTiD,cAsTvB,CAAe,OAAf,CAA1B,C;;QAtThB,OAwTO,W;O;KAnUX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAuTL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAxToC,WAwThC,CAAY,OAAZ,CAAJ,EAxTiD,cAwTvB,CAAe,OAAf,CAA1B,C;;QAxThB,OA0TO,W;O;KArUX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAyTL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA1ToC,WA0ThC,CAAY,OAAZ,CAAJ,EA1TiD,cA0TvB,CAAe,OAAf,CAA1B,C;;QA1ThB,OA4TO,W;O;KAvUX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA2TL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA5ToC,WA4ThC,CAAY,OAAZ,CAAJ,EA5TiD,cA4TvB,CAAe,OAAf,CAA1B,C;;QA5ThB,OA8TO,W;O;KAzUX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA6TL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA9ToC,WA8ThC,CAAY,OAAZ,CAAJ,EA9TiD,cA8TvB,CAAe,OAAf,CAA1B,C;;QA9ThB,OAgUO,W;O;KA3UX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA+TL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAhUoC,WAgUhC,CAAY,OAAZ,CAAJ,EAhUiD,cAgUvB,CAAe,OAAf,CAA1B,C;;QAhUhB,OAkUO,W;O;KA7UX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAiUL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAlUoC,WAkUhC,CAAY,OAAZ,CAAJ,EAlUiD,cAkUvB,CAAe,OAAf,CAA1B,C;;QAlUhB,OAoUO,W;O;KA/UX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAmUL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aApUoC,WAoUhC,CAAY,OAAZ,CAAJ,EApUiD,cAoUvB,CAAe,OAAf,CAA1B,C;;QApUhB,OAsUO,W;O;KAjVX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAqUL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAtUoC,WAsUhC,CAAY,OAAZ,CAAJ,EAtUiD,cAsUvB,CAAe,OAAf,CAA1B,C;;QAtUhB,OAwUO,W;O;KAnVX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAuUL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAxUoC,WAwUhC,CAAY,OAAZ,CAAJ,EAxUiD,cAwUvB,CAAe,OAAf,CAA1B,C;;QAxUhB,OA0UO,W;O;KArVX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAyUL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA1UoC,WA0UhC,CAAY,OAAZ,CAAJ,EA1UiD,cA0UvB,CAAe,OAAf,CAA1B,C;;QA1UhB,OA4UO,W;O;KAvVX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA2UL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA5UoC,WA4UhC,CAAY,OAAZ,CAAJ,EA5UiD,cA4UvB,CAAe,OAAf,CAA1B,C;;QA5UhB,OA8UO,W;O;KAzVX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA6UL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA9UoC,WA8UhC,CAAY,OAAZ,CAAJ,EA9UiD,cA8UvB,CAAe,OAAf,CAA1B,C;;QA9UhB,OAgVO,W;O;KA3VX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA+UL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAhVoC,WAgVhC,CAAY,OAAZ,CAAJ,EAhViD,cAgVvB,CAAe,OAAf,CAA1B,C;;QAhVhB,OAkVO,W;O;KA7VX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAiVL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAlVoC,WAkVhC,CAAY,OAAZ,CAAJ,EAlViD,cAkVvB,CAAe,OAAf,CAA1B,C;;QAlVhB,OAoVO,W;O;KA/VX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAmVL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aApVoC,WAoVhC,CAAY,OAAZ,CAAJ,EApViD,cAoVvB,CAAe,OAAf,CAA1B,C;;QApVhB,OAsVO,W;O;KAjWX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAqVL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAtVoC,WAsVhC,CAAY,OAAZ,CAAJ,EAtViD,cAsVvB,CAAe,OAAf,CAA1B,C;;QAtVhB,OAwVO,W;O;KAnWX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAuVL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAxVoC,WAwVhC,CAAY,OAAZ,CAAJ,EAxViD,cAwVvB,CAAe,OAAf,CAA1B,C;;QAxVhB,OA0VO,W;O;KArWX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAyVL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA1VoC,WA0VhC,CAAY,OAAZ,CAAJ,EA1ViD,cA0VvB,CAAe,OAAf,CAA1B,C;;QA1VhB,OA4VO,W;O;KAvWX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA2VL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA5VoC,WA4VhC,CAAY,OAAZ,CAAJ,EA5ViD,cA4VvB,CAAe,OAAf,CAA1B,C;;QA5VhB,OA8VO,W;O;KAzWX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA6VL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA9VoC,WA8VhC,CAAY,OAAZ,CAAJ,EA9ViD,cA8VvB,CAAe,OAAf,CAA1B,C;;QA9VhB,OAgWO,W;O;KA3WX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA+VL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAhWoC,WAgWhC,CAAY,OAAZ,CAAJ,EAhWiD,cAgWvB,CAAe,OAAf,CAA1B,C;;QAhWhB,OAkWO,W;O;KA7WX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAiWL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAlWoC,WAkWhC,CAAY,OAAZ,CAAJ,EAlWiD,cAkWvB,CAAe,OAAf,CAA1B,C;;QAlWhB,OAoWO,W;O;KA/WX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAmWL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aApWoC,WAoWhC,CAAY,OAAZ,CAAJ,EApWiD,cAoWvB,CAAe,OAAf,CAA1B,C;;QApWhB,OAsWO,W;O;KAjXX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAqWL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAtWoC,WAsWhC,CAAY,OAAZ,CAAJ,EAtWiD,cAsWvB,CAAe,OAAf,CAA1B,C;;QAtWhB,OAwWO,W;O;KAnXX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAuWL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAxWoC,WAwWhC,CAAY,OAAZ,CAAJ,EAxWiD,cAwWvB,CAAe,OAAf,CAA1B,C;;QAxWhB,OA0WO,W;O;KArXX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAyWL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA1WoC,WA0WhC,CAAY,OAAZ,CAAJ,EA1WiD,cA0WvB,CAAe,OAAf,CAA1B,C;;QA1WhB,OA4WO,W;O;KAvXX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA2WL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA5WoC,WA4WhC,CAAY,OAAZ,CAAJ,EA5WiD,cA4WvB,CAAe,OAAf,CAA1B,C;;QA5WhB,OA8WO,W;O;KAzXX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA6WL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA9WoC,WA8WhC,CAAY,OAAZ,CAAJ,EA9WiD,cA8WvB,CAAe,OAAf,CAA1B,C;;QA9WhB,OAgXO,W;O;KA3XX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA+WL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAhXoC,WAgXhC,CAAY,OAAZ,CAAJ,EAhXiD,cAgXvB,CAAe,OAAf,CAA1B,C;;QAhXhB,OAkXO,W;O;KA7XX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAiXL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAlXoC,WAkXhC,CAAY,OAAZ,CAAJ,EAlXiD,cAkXvB,CAAe,OAAf,CAA1B,C;;QAlXhB,OAoXO,W;O;KA/XX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAmXL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aApXoC,WAoXhC,CAAY,OAAZ,CAAJ,EApXiD,cAoXvB,CAAe,OAAf,CAA1B,C;;QApXhB,OAsXO,W;O;KAjYX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAqXL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAtXoC,WAsXhC,CAAY,OAAZ,CAAJ,EAtXiD,cAsXvB,CAAe,OAAf,CAA1B,C;;QAtXhB,OAwXO,W;O;KAnYX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAuXL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAxXoC,WAwXhC,CAAY,OAAZ,CAAJ,EAxXiD,cAwXvB,CAAe,OAAf,CAA1B,C;;QAxXhB,OA0XO,W;O;KArYX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAyXL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA1XoC,WA0XhC,CAAY,OAAZ,CAAJ,EA1XiD,cA0XvB,CAAe,OAAf,CAA1B,C;;QA1XhB,OA4XO,W;O;KAvYX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA2XL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA5XoC,WA4XhC,CAAY,OAAZ,CAAJ,EA5XiD,cA4XvB,CAAe,OAAf,CAA1B,C;;QA5XhB,OA8XO,W;O;KAzYX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA6XL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA9XoC,WA8XhC,CAAY,OAAZ,CAAJ,EA9XiD,cA8XvB,CAAe,OAAf,CAA1B,C;;QA9XhB,OAgYO,W;O;KA3YX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA+XL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAhYoC,WAgYhC,CAAY,OAAZ,CAAJ,EAhYiD,cAgYvB,CAAe,OAAf,CAA1B,C;;QAhYhB,OAkYO,W;O;KA7YX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAiYL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAlYoC,WAkYhC,CAAY,OAAZ,CAAJ,EAlYiD,cAkYvB,CAAe,OAAf,CAA1B,C;;QAlYhB,OAoYO,W;O;KA/YX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAmYL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aApYoC,WAoYhC,CAAY,OAAZ,CAAJ,EApYiD,cAoYvB,CAAe,OAAf,CAA1B,C;;QApYhB,OAsYO,W;O;KAjZX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAqYL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAtYoC,WAsYhC,CAAY,OAAZ,CAAJ,EAtYiD,cAsYvB,CAAe,OAAf,CAA1B,C;;QAtYhB,OAwYO,W;O;KAnZX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAuYL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aAxYoC,WAwYhC,CAAY,OAAZ,CAAJ,EAxYiD,cAwYvB,CAAe,OAAf,CAA1B,C;;QAxYhB,OA0YO,W;O;KArZX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QAyYL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA1YoC,WA0YhC,CAAY,OAAZ,CAAJ,EA1YiD,cA0YvB,CAAe,OAAf,CAA1B,C;;QA1YhB,OA4YO,W;O;KAvZX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA2YL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA5YoC,WA4YhC,CAAY,OAAZ,CAAJ,EA5YiD,cA4YvB,CAAe,OAAf,CAA1B,C;;QA5YhB,OA8YO,W;O;KAzZX,C;+FAcA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAUI,eAAiC,cAAlB,YAAY,gBAAZ,CAAkB,EAAc,EAAd,C;QAC1B,kBAAc,mBAAoB,QAAApB,C;QA6YL,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAAY,aA9YoC

MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAY,aAAI,YAAY,OAAZ,CAAJ,EAA0B,eAAe,OAAf,CAA1B,C;;MAEhB,OAAO,W;K;mGAGX,+D;MAUoB,Q;

oBAAJ,EAtJe,aAsJF,CAAc,oBAAd,CAAb,C;;QAtJhB,OAAuB,M;O;KAd3B,C;oGAiBA,iD;MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAY,aAAI,OAAJ,EAAa,cAAc,OAAd,CAAb,C;;MAEhB,OAAO,W;K;sGAGX,iD;MAW

a,WAAA,SAAb,M;QAAmB,IAAK,WAAI,IAAJ,C;;MACxB,OAAO,I;K;IAGX,oC;MAKiB,Q;MADb,WAAW,iBA
Ae,gBAAf,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAmB,IAAK,WAAI,IAAJ,C;;MACxB,OAAO,
I;K;IAGX,oC;MAKiB,Q;MADb,WAAW,iBAAgB,gBAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M
;QAAmB,IAAK,WAAI,IAAJ,C;;MACxB,OAAO,I;K;IAGX,oC;MAKiB,Q;MADb,WAAW,iBAAiB,gBAAjB,C;M
ACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAmB,IAAK,WAAI,IAAJ,C;;MACxB,OAAO,I;K;IAGX,oC;
MAKiB,Q;MADb,WAAW,iBAAkB,gBAAlB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAmB,IAAK
K,WAAI,IAAJ,C;;MACxB,OAAO,I;K;IAGX,oC;MAKiB,Q;MADb,WAAW,iBAAmB,gBAAnB,C;MACX,wBAA
a,SAAb,gB;QAAa,WAAA,SAAb,M;QAAmB,IAAK,WAAI,IAAJ,C;;MACxB,OAAO,I;K;IAGX,oC;MAKiB,Q;M
ADb,WAAW,iBAAgB,gBAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAb,UAAa,SAAb,O;QAAmB,IAAK,WA
AI,iBAAJ,C;;MACxB,OAAO,I;K;IAGX,0B;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAA
L,K;aACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;;UACQ,+BAAa,qBAAiB,YAAY,gBAAZ,CAAjB,
CAAb,C;UAHL,K;;MAAP,W;K;IAOJ,4B;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,
K;aACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;;UACQ,iCAAa,qBAAoB,YAAY,gBAAZ,CAApB,C
AAb,C;UAHL,K;;MAAP,W;K;IAOJ,4B;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K
;aACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;;UACQ,iCAAa,qBAAqB,YAAY,gBAAZ,CAArB,CA
Ab,C;UAHL,K;;MAAP,W;K;IAOJ,4B;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K;a
ACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;;UACQ,iCAAa,qBAAmB,YAAY,gBAAZ,CAAnB,CAA
b,C;UAHL,K;;MAAP,W;K;IAOJ,4B;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K;aA
CA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;;UACQ,iCAAa,qBAAoB,YAAY,gBAAZ,CApB,CAAb,
C;UAHL,K;;MAAP,W;K;IAOJ,4B;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K;aAC
A,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;;UACQ,iCAAa,qBAAqB,YAAY,gBAAZ,CArB,CAb,C;
UAHL,K;;MAAP,W;K;IAOJ,4B;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K;aACA,
C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;;UACQ,iCAAa,qBAAsB,YAAY,gBAAZ,CAtB,CAb,C;U
AHL,K;;MAAP,W;K;IAOJ,4B;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K;aACA,C
;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;;UACQ,iCAAa,qBAAuB,YAAY,gBAAZ,CAvB,CAb,C;UA
HL,K;;MAAP,W;K;IAOJ,4B;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K;aACA,C;
UAAK,aAAM,sBAAK,CAAL,EAAN,C;UAAL,K;;UACQ,iCAAa,qBAAoB,YAAiB,eAAL,gBAAK,GAAL,Gb,
CAAjB,CAApB,CAAb,C;UAHL,K;;MAAP,W;K;oFAOJ,yB;MAAA,+D;MAAwaA,gD;MAAxaA,uC;QAMW,kBAA
U,gB;QAAsD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAva6B,SAualB,CAAU,OAAV,C;UACC,OAAZ,WAAY,E
AAO,IAAP,C;;QAAxahB,OA0aO,W;O;KAhbX,C;sFASA,yB;MAAA,+D;MA0aA,gD;MA1aA,uC;QAMW,kBAAU,
gB;QAAwaD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAza6B,SAyalB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EA
AO,IAAP,C;;QA1ahB,OA4aO,W;O;KAlbX,C;sFASA,yB;MAAA,+D;MA4aA,gD;MA5aA,uC;QAMW,kBAAU,g
B;QA0aD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA3a6B,SA2alB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAA
O,IAAP,C;;QA5ahB,OA8aO,W;O;KApbX,C;sFASA,yB;MAAA,+D;MA8aA,gD;MA9aA,uC;QAMW,kBAAU,gB;
QA4aD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA7a6B,SA6alB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,
IAAP,C;;QA9ahB,OAgbO,W;O;KAtbX,C;sFASA,yB;MAAA,+D;MAgbA,gD;MAhbA,uC;QAMW,kBAAU,gB;Q
A8aD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA/a6B,SA+alB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IA
AP,C;;QAhbhB,OAkbO,W;O;KAxbX,C;sFASA,yB;MAAA,+D;MAkbA,gD;MAlbA,uC;QAMW,kBAAU,gB;QAg
bD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAjb6B,SAiblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP
,C;;QAlbhB,OAobO,W;O;KA1bX,C;sFASA,yB;MAAA,+D;MAobA,gD;MApbA,uC;QAMW,kBAAU,gB;QAkbD,
Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAnb6B,SAmblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;
;QApbhB,OAsbO,W;O;KA5bX,C;sFASA,yB;MAAA,+D;MAsbA,gD;MAtbA,uC;QAMW,kBAAU,gB;QAobD,Q;
QAAhB,iD;UAAgB,cAAhB,e;UACI,WArb6B,SAqblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;Q
AtbhB,OAwbO,W;O;KA9bX,C;sFASA,yB;MAAA,+D;MAwbA,oC;MAAA,gD;MAAA,gC;MAAxbA,uC;QAMW,kBAAU
,gB;QAsbD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAvb6B,SAublB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,
IAAP,C;;QAxbhB,OA0bO,W;O;KAhcX,C;sFASA,yB;MAAA,+D;MA0bA,gD;MA1bA,uC;QAMW,kBAAU,gB;QAwb
D,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAzb6B,SAyblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;
QA1bhB,OA4bO,W;O;KAlcX,C;sFASA,yB;MAAA,+D;MA4bA,gD;MA5bA,uC;QAMW,kBAAU,gB;QA0bD,Q;Q
AAhB,iD;UAAgB,cAAhB,e;UACI,WA3b6B,SA2blB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5
bhB,OA8bO,W;O;KApcX,C;sFASA,yB;MAAA,+D;MA8bA,gD;MA9bA,uC;QAMW,kBAAU,gB;QA4bD,Q;QAAh
B,iD;UAAgB,cAAhB,e;UACI,WA7b6B,SA6blB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9bhB
,OAgcO,W;O;KAtcX,C;sFASA,yB;MAAA,+D;MAgcA,gD;MAhcA,uC;QAMW,kBAAU,gB;QA8bD,Q;QAAhB,iD
;UAAgB,cAAhB,e;UACI,WA/b6B,SA+blB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhchB,OAk
cO,W;O;KAxcX,C;sFASA,yB;MAAA,+D;MAkcA,gD;MAlcA,uC;QAMW,kBAAU,gB;QAgcD,Q;QAAhB,iD;UA
Ag,cAAhB,e;UACI,WAjc6B,SAicblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlchB,OAocO,W;
O;KA1cX,C;sFASA,yB;MAAA,+D;MAocA,gD;MApcA,uC;QAMW,kBAAU,gB;QAkcD,Q;QAAhB,iD;UAAgB,
cAAhB,e;UACI,WAnc6B,SAmcblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApchB,OAscO,W;O;
KA5cX,C;sFASA,yB;MAAA,+D;MAscA,gD;MAtcA,uC;QAMW,kBAAU,gB;QAocD,Q;QAAhB,iD;UAAgB,cAAh
B,e;UACI,WArc6B,SAqcblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtchB,OAwcO,W;O;KA9
cX,C;sFASA,yB;MAAA,+D;MAwcA,gD;MAxcA,uC;QAMW,kBAAU,gB;QAscD,Q;QAAhB,iD;UAAgB,cAAhB,
e;UACI,WAvc6B,SAucblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxchB,OA0cO,W;O;KAhdX,
C;sFASA,yB;MAAA,+D;MA0cA,gD;MA1cA,uC;QAMW,kBAAU,gB;QAwcD,Q;QAAhB,iD;UAAgB,cAAhB,e;UA
CI,WAzc6B,SAycblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1chB,OA4cO,W;O;KAldX,C;sF
ASA,yB;MAAA,+D;MA4cA,gD;MA5cA,uC;QAMW,kBAAU,gB;QA0cD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,W
A3c6B,SA2cblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5chB,OA8cO,W;O;KApdX,C;sFASA,
yB;MAAA,+D;MA8cA,gD;MA9cA,uC;QAMW,kBAAU,gB;QA4cD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA7c6
B,SA6cblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9chB,OAgdO,W;O;KAtdX,C;sFASA,yB;MA
Ag+D;MAhdA,uC;QAMW,kBAAU,gB;QA8cD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA/c6B,SA+cblB,CAAU,
OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhdhB,OAkdO,W;O;KAxdX,C;sFASA,yB;MAAA,+D;MAkdA,gD
;MAldA,uC;QAMW,kBAAU,gB;QAgdD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAjd6B,SAidblB,CAAU,OAAV,C
;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAldhB,OAodO,W;O;KA1dX,C;sFASA,yB;MAAA,+D;MAodA,gD;MApdA,
uC;QAMW,kBAAU,gB;QAkdD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAnd6B,SAmdblB,CAAU,OAAV,C;UACC
,OAAZ,WAAY,EAAO,IAAP,C;;QApdhB,OAsdO,W;O;KA5dX,C;sFASA,yB;MAAA,+D;MAsdA,gD;MAtdA,uC;Q
AMW,kBAAU,gB;QAodD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WArd6B,SAqdblB,CAAU,OAAV,C;UACC,OAAZ
,WAAY,EAAO,IAAP,C;;QAtdhB,OAwdO,W;O;KA9dX,C;sFASA,yB;MAAA,+D;MAwdA,gD;MAxdA,uC;QAMW
,kBAAU,gB;QAsdD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAvd6B,SAudblB,CAAU,OAAV,C;UACC,OAAZ,WA
AY,EAAO,IAAP,C;;QAxdhB,OA0dO,W;O;KAheX,C;sFASA,yB;MAAA,+D;MA0dA,gD;MA1dA,uC;QAMW,kBA
AU,gB;QAwdD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAzd6B,SAydblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,
EAAO,IAAP,C;;QA1dhB,OA4dO,W;O;KAleX,C;sFASA,yB;MAAA,+D;MA4dA,gD;MA5dA,uC;QAMW,kBAAU,
gB;QA0dD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA3d6B,SA2dblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO
,IAAP,C;;QA5dhB,OA8dO,W;O;KApeX,C;sFASA,yB;MAAA,+D;MA8dA,gD;MA9dA,uC;QAMW,kBAAU,gB;Q
A4dD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA7d6B,SA6dblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAA
P,C;;QA9dhB,OAgeO,W;O;KAteX,C;sFASA,yB;MAAA,+D;MAgeA,gD;MAheA,uC;QAMW,kBAAU,gB;QA8dD,
Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA/d6B,SA+dblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;
QAheA,OAkeO,W;O;KAxeX,C;sFASA,yB;MAAA,+D;MAkeA,gD;MAleA,uC;QAMW,kBAAU,gB;QAgeD,Q;QAA
hB,iD;UAAgB,cAAhB,e;UACI,WAje6B,SAiedlB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlehB
,OAoeO,W;O;KA1eX,C;sFASA,yB;MAAA,+D;MAoeA,gD;MApeA,uC;QAMW,kBAAU,gB;QAkeD,Q;QAAhB,iD
;UAAgB,cAAhB,e;UACI,WAne6B,SAmedlB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApehB,OAs
eO,W;O;KA5eX,C;sFASA,yB;MAAA,+D;MAseA,gD;MAteA,uC;QAMW,kBAAU,gB;QAoeD,Q;QAAhB,iD;UAAg
B,cAAhB,e;UACI,WAre6B,SAqedlB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtehB,OAweO,W;O;
KA9eX,C;sFASA,yB;MAAA,+D;MAweA,gD;MAxeA,uC;QAMW,kBAAU,gB;QAseD,Q;QAAhB,iD;UAAgB,cAAh
B,e;UACI,WAve6B,SAuedlB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxehB,OA0eO,W;O;KAhf
X,C;sFASA,yB;MAAA,+D;MA0eA,gD;MA1eA,uC;QAMW,kBAAU,gB;QAweD,Q;QAAhB,iD;UAAgB,cAAhB,e;UA
CI,WAze6B,SAyedlB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1ehB,OA4eO,W;O;KAlfX,C;sF
ASA,yB;MAAA,+D;MA4eA,gD;MA5eA,uC;QAMW,kBAAU,gB;QA0eD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,W
A3e6B,SA2edlB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5ehB,OA8eO,W;O;KApfX,C;sFASA,
yB;MAAA,+D;MA8eA,gD;MA9eA,uC;QAMW,kBAAU,gB;QA4eD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA7e6
B,SA6edlB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9ehB,OAgfO,W;O;KAtfX,C;sFASA,yB;MA
Ag+D;MAhfA,uC;QAMW,kBAAU,gB;QA8eD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA/e6B,SA+eblB,CAAU,
OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhfhB,OAkfO,W;O;KAxfX,C;sFASA,yB;MAAA,+D;MAkfA,gD
;MAlfA,uC;QAMW,kBAAU,gB;QAgfD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAjf6B,SAifblB,CAAU,OAAV,C
;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlfhB,OAofO,W;O;KA1fX,C;sFASA,yB;MAAA,+D;MAofA,gD;MApfA,
uC;QAMW,kBAAU,gB;QAkfD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAnf6B,SAmfblB,CAAU,OAAV,C;UACC
,OAAZ,WAAY,EAAO,IAAP,C;;QApfhB,OAsfO,W;O;KA5fX,C;sFASA,yB;MAAA,+D;MAsfA,gD;MAtfA,uC;Q
AMW,kBAAU,gB;QAofD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WArf6B,SAqfblB,CAAU,OAAV,C;UACC,OAAZ
,WAAY,EAAO,IAAP,C;;QAtfhB,OAwfO,W;O;KA9fX,C;sFASA,yB;MAAA,+D;MAwfA,gD;MAxfA,uC;QAMW
,kBAAU,gB;QAsfD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAvf6B,SAufblB,CAAU,OAAV,C;UACC,OAAZ,WA
AY,EAAO,IAAP,C;;QAxfhB,OA0fO,W;O;KAhgBX,C;sFASA,yB;MAAA,+D;MA0fA,gD;MA1fA,uC;QAMW,kBA
AU,gB;QAwfD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAzf6B,SAyfblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,
EAAO,IAAP,C;;QA1fhB,OA4fO,W;O;KAlgBX,C;sFASA,yB;MAAA,+D;MA4fA,gD;MA5fA,uC;QAMW,kBAAU,
gB;QA0fD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA3f6B,SA2fblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO
,IAAP,C;;QA5fhB,OA8fO,W;O;KApgBX,C;sFASA,yB;MAAA,+D;MA8fA,gD;MA9fA,uC;QAMW,kBAAU,gB;Q
A4fD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA7f6B,SA6fblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAA
P,C;;QA9fhB,OAggBO,W;O;KAtgBX,C;sFASA,yB;MAAA,+D;MAggBA,gD;MAhgBA,uC;QAMW,kBAAU,gB;QA8fD,
Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA/f6B,SA+fblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;
QAhgBhB,OAkgBO,W;O;KAxgBX,C;sFASA,yB;MAAA,+D;MAkgBA,gD;MAlgBA,uC;QAMW,kBAAU,gB;QAggBD,Q;QAA
hB,iD;UAAgB,cAAhB,e;UACI,WAjgB6B,SAigBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlgBh
B,OAogBO,W;O;KA1gBX,C;sFASA,yB;MAAA,+D;MAogBA,gD;MApgBA,uC;QAMW,kBAAU,gB;QAkgBD,Q;QAAhB,iD
;UAAgB,cAAhB,e;UACI,WAngB6B,SAmgBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApgBhB,OAs
gBO,W;O;KA5gBX,C;sFASA,yB;MAAA,+D;MAsgBA,gD;MAtgBA,uC;QAMW,kBAAU,gB;QAogBD,Q;QAAhB,iD;UAAg
B,cAAhB,e;UACI,WArgB6B,SAqgBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtgBhB,OAwgBO,W;O;
KA9gBX,C;sFASA,yB;MAAA,+D;MAwgBA,gD;MAxgBA,uC;QAMW,kBAAU,gB;QAsgBD,Q;QAAhB,iD;UAAgB,cAAh
B,e;UACI,WAvgB6B,SAugBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxgBhB,OA0gBO,W;O;KAhh
BX,C;sFASA,yB;MAAA,+D;MA0gBA,gD;MA1gBA,uC;QAMW,kBAAU,gB;QAwgBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UA
CI,WAzgB6B,SAygBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1gBhB,OA4gBO,W;O;KAlhBX,C;sF
ASA,yB;MAAA,+D;MA4gBA,gD;MA5gBA,uC;QAMW,kBAAU,gB;QA0gBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,W
A3gB6B,SA2gBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5gBhB,OA8gBO,W;O;KAphBX,C;sFASA,
yB;MAAA,+D;MA8gBA,gD;MA9gBA,uC;QAMW,kBAAU,gB;QA4gBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA7gB6
B,SA6gBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9gBhB,OAghBO,W;O;KAthBX,C;sFASA,yB;MA
Ag+D;MAhhBA,uC;QAMW,kBAAU,gB;QA8gBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA/gB6B,SA+gBblB,CAAU,
OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhhBhB,OAkhBO,W;O;KAxhBX,C;sFASA,yB;MAAA,+D;MAkhBA,gD
;MAlhBA,uC;QAMW,kBAAU,gB;QAghBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAjhB6B,SAihBblB,CAAU,OAAV,C
;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlhBhB,OAohBO,W;O;KA1hBX,C;sFASA,yB;MAAA,+D;MAohBA,gD;MAphBA,
uC;QAMW,kBAAU,gB;QAkhBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAnhB6B,SAmhBblB,CAAU,OAAV,C;UACC
,OAAZ,WAAY,EAAO,IAAP,C;;QAphBhB,OAshBO,W;O;KA5hBX,C;sFASA,yB;MAAA,+D;MAshBA,gD;MAthBA,uC;Q
AMW,kBAAU,gB;QAohBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WArhB6B,SAqhBblB,CAAU,OAAV,C;UACC,OAAZ
,WAAY,EAAO,IAAP,C;;QAthBhB,OAwhBO,W;O;KA9hBX,C;sFASA,yB;MAAA,+D;MAwhBA,gD;MAxhBA,uC;QAMW
,kBAAU,gB;QAshBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAvhB6B,SAuhBblB,CAAU,OAAV,C;UACC,OAAZ,WA
AY,EAAO,IAAP,C;;QAxhBhB,OA0hBO,W;O;KAhiBX,C;sFASA,yB;MAAA,+D;MA0hBA,gD;MA1hBA,uC;QAMW,kBA
AU,gB;QAwhBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAzhB6B,SAyhBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,
EAAO,IAAP,C;;QA1hBhB,OA4hBO,W;O;KAliBX,C;sFASA,yB;MAAA,+D;MA4hBA,gD;MA5hBA,uC;QAMW,kBAAU,
gB;QA0hBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA3hB6B,SA2hBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO
,IAAP,C;;QA5hBhB,OA8hBO,W;O;KApiBX,C;sFASA,yB;MAAA,+D;MA8hBA,gD;MA9hBA,uC;QAMW,kBAAU,gB;Q
A4hBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA7hB6B,SA6hBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAA
P,C;;QA9hBhB,OAgiBO,W;O;KAtiBX,C;sFASA,yB;MAAA,+D;MAgiBA,gD;MAhiBA,uC;QAMW,kBAAU,gB;QA8hBD,
Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA/hB6B,SA+hBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;
QAhiBhB,OAkiBO,W;O;KAxiBX,C;sFASA,yB;MAAA,+D;MAkiBA,gD;MAliBA,uC;QAMW,kBAAU,gB;QAgiBD,Q;QAA
hB,iD;UAAgB,cAAhB,e;UACI,WAjiB6B,SAiiBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAliBh
B,OAoiBO,W;O;KA1iBX,C;sFASA,yB;MAAA,+D;MAoiBA,gD;MApiBA,uC;QAMW,kBAAU,gB;QAkiBD,Q;QAAhB,iD
;UAAgB,cAAhB,e;UACI,WAniB6B,SAmiBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApiBhB,OAs
iBO,W;O;KA5iBX,C;sFASA,yB;MAAA,+D;MAsiBA,gD;MAtiBA,uC;QAMW,kBAAU,gB;QAoiBD,Q;QAAhB,iD;UAAg
B,cAAhB,e;UACI,WAriB6B,SAqiBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtiBhB,OAwiBO,W;O;
KA9iBX,C;sFASA,yB;MAAA,+D;MAwiBA,gD;MAxiBA,uC;QAMW,kBAAU,gB;QAsiBD,Q;QAAhB,iD;UAAgB,cAAh
B,e;UACI,WAviB6B,SAuiBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxiBhB,OA0iBO,W;O;KAhj
BX,C;sFASA,yB;MAAA,+D;MA0iBA,gD;MA1iBA,uC;QAMW,kBAAU,gB;QAwiBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UA
CI,WAziB6B,SAyiBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1iBhB,OA4iBO,W;O;KAljBX,C;sF
ASA,yB;MAAA,+D;MA4iBA,gD;MA5iBA,uC;QAMW,kBAAU,gB;QA0iBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,W
A3iB6B,SA2iBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5iBhB,OA8iBO,W;O;KApjBX,C;sFASA,
yB;MAAA,+D;MA8iBA,gD;MA9iBA,uC;QAMW,kBAAU,gB;QA4iBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA7iB6
B,SA6iBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9iBhB,OAgjBO,W;O;KAtjBX,C;sFASA,yB;MA
Ag+D;MAhjBA,uC;QAMW,kBAAU,gB;QA8iBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA/iB6B,SA+iBblB,CAAU,
OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhjBhB,OAkjBO,W;O;KAxjBX,C;sFASA,yB;MAAA,+D;MAkjBA,gD
;MAljBA,uC;QAMW,kBAAU,gB;QAgjBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAjjB6B,SAijBblB,CAAU,OAAV,C
;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAljBhB,OAojBO,W;O;KA1jBX,C;sFASA,yB;MAAA,+D;MAojBA,gD;MApjBA,
uC;QAMW,kBAAU,gB;QAkjBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAnjB6B,SAmjBblB,CAAU,OAAV,C;UACC
,OAAZ,WAAY,EAAO,IAAP,C;;QApjBhB,OAsjBO,W;O;KA5jBX,C;sFASA,yB;MAAA,+D;MAsjBA,gD;MAtjBA,uC;Q
AMW,kBAAU,gB;QAojBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WArjB6B,SAqjBblB,CAAU,OAAV,C;UACC,OAAZ
,WAAY,EAAO,IAAP,C;;QAtjBhB,OAwjBO,W;O;KA9jBX,C;sFASA,yB;MAAA,+D;MAwjBA,gD;MAxjBA,uC;QAMW
,kBAAU,gB;QAsjBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAvjB6B,SAujBblB,CAAU,OAAV,C;UACC,OAAZ,WA
AY,EAAO,IAAP,C;;QAxjBhB,OA0jBO,W;O;KAhkBX,C;sFASA,yB;MAAA,+D;MA0jBA,gD;MA1jBA,uC;QAMW,kBA
AU,gB;QAwjBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAzjB6B,SAyjBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,
EAAO,IAAP,C;;QA1jBhB,OA4jBO,W;O;KAlkBX,C;sFASA,yB;MAAA,+D;MA4jBA,gD;MA5jBA,uC;QAMW,kBAAU,
gB;QA0jBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA3jB6B,SA2jBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO
,IAAP,C;;QA5jBhB,OA8jBO,W;O;KApkBX,C;sFASA,yB;MAAA,+D;MA8jBA,gD;MA9jBA,uC;QAMW,kBAAU,gB;Q
A4jBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA7jB6B,SA6jBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAA
P,C;;QA9jBhB,OAgkBO,W;O;KAtkBX,C;sFASA,yB;MAAA,+D;MAgkBA,gD;MAhkBA,uC;QAMW,kBAAU,gB;QA8jBD,
Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA/jB6B,SA+jBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;
QAhkBhB,OAkkBO,W;O;KAxkBX,C;sFASA,yB;MAAA,+D;MAkkBA,gD;MAlkBA,uC;QAMW,kBAAU,gB;QAgkBD,Q;QAA
hB,iD;UAAgB,cAAhB,e;UACI,WAjkB6B,SAikBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlkBh
B,OAokBO,W;O;KA1kBX,C;sFASA,yB;MAAA,+D;MAokBA,gD;MApkBA,uC;QAMW,kBAAU,gB;QAkkBD,Q;QAAhB,iD
;UAAgB,cAAhB,e;UACI,WAnkB6B,SAmkBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApkBhB,OAs
kBO,W;O;KA5kBX,C;sFASA,yB;MAAA,+D;MAskBA,gD;MAtkBA,uC;QAMW,kBAAU,gB;QAokBD,Q;QAAhB,iD;UAAg
B,cAAhB,e;UACI,WArkB6B,SAqkBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtkBhB,OAwkBO,W;O;
KA9kBX,C;sFASA,yB;MAAA,+D;MAwkBA,gD;MAxkBA,uC;QAMW,kBAAU,gB;QAskBD,Q;QAAhB,iD;UAAgB,cAAh
B,e;UACI,WAvkB6B,SAukBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxkBhB,OA0kBO,W;O;KAhl
BX,C;sFASA,yB;MAAA,+D;MA0kBA,gD;MA1kBA,uC;QAMW,kBAAU,gB;QAwkBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UA
CI,WAzkB6B,SAykBblB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1kBhB,OA4kBO,W;O;KAllBX,C;sF
ASA,yB;MAAA,+D;MA4kBA,gD;MA5kBA,uC;QAMW,kBAAU,gB;QA0k

AAiB,gB;QAqJR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAtJoC,SAsJzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAvJhB,OAyJO,W;O;KArKX,C;oGAeA,yB;MAAA,+D;MAyJA,gD;MAzJA,uC;QAAY,kBAAiB,gB;QAwJR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAzJoC,SAyJzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1JhB,OA4JO,W;O;KAxKX,C;oGAeA,yB;MAAA,+D;MA4JA,gD;MA5JA,uC;QAAY,kBAAiB,gB;QA2JR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA5JoC,SA4JzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7JhB,OA+JO,W;O;KA3KX,C;oGAeA,yB;MAAA,+D;MA+JA,gD;MA/JA,uC;QAAY,kBAAiB,gB;QA8JR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA/JoC,SA+JzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhKhB,OAkKO,W;O;KA9KX,C;oGAeA,yB;MAAA,+D;MAkKA,gD;MAlKA,uC;QAAY,kBAAiB,gB;QAiKR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAlKoC,SAkKzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAnKhB,OAqKO,W;O;KAjLX,C;oGAeA,yB;MAAA,+D;MAqKA,gD;MArKA,uC;QAAY,kBAAiB,gB;QAoKR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WArKoC,SAqKzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtKhB,OAwKO,W;O;KApLX,C;oGAeA,yB;MAAA,+D;MAwKA,gD;MAxKA,uC;QAAY,kBAAiB,gB;QAuKR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAxKoC,SAwKzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAzKhB,OA2KO,W;O;KAvLX,C;oGAeA,yB;MAAA,+D;MA2KA,gD;MA3KA,uC;QAAY,kBAAiB,gB;QA0KR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA3KoC,SA2KzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5KhB,OA8KO,W;O;KA1LX,C;oGAeA,yB;MAAA,+D;MA8KA,gD;MA9KA,uC;QAAY,kBAAiB,gB;QA6KR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA9KoC,SA8KzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/KhB,OAiLO,W;O;KA7LX,C;oGAeA,yB;MAAA,+D;MAiLA,gD;MAjLA,uC;QAAY,kBAAiB,gB;QAgLR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAjLoC,SAiLzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlLhB,OAoLO,W;O;KAhMX,C;oGAeA,yB;MAAA,+D;MAoLA,gD;MApLA,uC;QAAY,kBAAiB,gB;QAmLR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WApLoC,SAoLzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QArLhB,OAuLO,W;O;KAnMX,C;oGAeA,yB;MAAA,+D;MAuLA,gD;MAvLA,uC;QAAY,kBAAiB,gB;QAsLR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAvLoC,SAuLzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxLhB,OA0LO,W;O;KAtMX,C;oGAeA,yB;MAAA,+D;MA0LA,gD;MA1LA,uC;QAAY,kBAAiB,gB;QAyLR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA1LoC,SA0LzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA3LhB,OA6LO,W;O;KAzMX,C;oGAeA,yB;MAAA,+D;MA6LA,gD;MA7LA,uC;QAAY,kBAAiB,gB;QA4LR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA7LoC,SA6LzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9LhB,OAgMO,W;O;KA5MX,C;oGAeA,yB;MAAA,+D;MAgMA,gD;MAhMA,uC;QAAY,kBAAiB,gB;QA+LR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAhMoC,SAgMzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAjMhB,OAmMO,W;O;KA/MX,C;oGAeA,yB;MAAA,+D;MAmMA,gD;MAnMA,uC;QAAY,kBAAiB,gB;QAkMR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAnMoC,SAmMzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApMhB,OAsMO,W;O;KAlNX,C;oGAeA,yB;MAAA,+D;MAsMA,gD;MAtMA,uC;QAAY,kBAAiB,gB;QAqMR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAtMoC,SAsMzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAvMhB,OAyMO,W;O;KArNX,C;oGAeA,yB;MAAA,+D;MAyMA,gD;MAzMA,uC;QAAY,kBAAiB,gB;QAwMR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAzMoC,SAyMzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1MhB,OA4MO,W;O;KAxNX,C;oGAeA,yB;MAAA,+D;MA4MA,gD;MA5MA,uC;QAAY,kBAAiB,gB;QA2MR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA5MoC,SA4MzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7MhB,OA+MO,W;O;KA3NX,C;oGAeA,yB;MAAA,+D;MA+MA,gD;MA/MA,uC;QAAY,kBAAiB,gB;QA8MR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA/MoC,SA+MzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhNhB,OAkNO,W;O;KA9NX,C;oGAeA,yB;MAAA,+D;MAkNA,gD;MAlNA,uC;QAAY,kBAAiB,gB;QAiNR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAlNoC,SAkNzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAnNhB,OAqNO,W;O;KAjOX,C;oGAeA,yB;MAAA,+D;MAqNA,gD;MArNA,uC;QAAY,kBAAiB,gB;QAoNR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WArNoC,SAqNzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtNhB,OAwNO,W;O;KApOX,C;oGAeA,yB;MAAA,+D;MAwNA,gD;MAxNA,uC;QAAY,kBAAiB,gB;QAuNR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAxNoC,SAwNzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAzNhB,OA2NO,W;O;KAvOX,C;oGAeA,yB;MAAA,+D;MA2NA,gD;MA3NA,uC;QAAY,kBAAiB,gB;QA0NR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA3NoC,SA2NzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5NhB,OA8NO,W;O;KA1OX,C;oGAeA,yB;MAAA,+D;MA8NA,gD;MA9NA,uC;QAAY,kBAAiB,gB;QA6NR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA9NoC,SA8NzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/NhB,OAiOO,W;O;KA7OX,C;oGAeA,yB;MAAA,+D;MAiOA,gD;MAjOA,uC;QAAY,kBAAiB,gB;QAgOR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAjOoC,SAiOzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlOhB,OAoOO,W;O;KAhPX,C;oGAeA,yB;MAAA,+D;MAoOA,gD;MApOA,uC;QAAY,kBAAiB,gB;QAmOR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WApOoC,SAoOzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QArOhB,OAuOO,W;O;KAnPX,C;oGAeA,yB;MAAA,+D;MAuOA,gD;MAvOA,uC;QAAY,kBAAiB,gB;QAsOR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAvOoC,SAuOzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxOhB,OA0OO,W;O;KAtPX,C;oGAeA,yB;MAAA,+D;MA0OA,gD;MA1OA,uC;QAAY,kBAAiB,gB;QAyOR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA1OoC,SA0OzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA3OhB,OA6OO,W;O;KAzPX,C;oGAeA,yB;MAAA,+D;MA6OA,gD;MA7OA,uC;QAAY,kBAAiB,gB;QA4OR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA7OoC,SA6OzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9OhB,OAgPO,W;O;KA5PX,C;oGAeA,yB;MAAA,+D;MAgPA,gD;MAhPA,uC;QAAY,kBAAiB,gB;QA+OR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAhPoC,SAgPzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAjPhB,OAmPO,W;O;KA/PX,C;oGAeA,yB;MAAA,+D;MAmPA,gD;MAnPA,uC;QAAY,kBAAiB,gB;QAkPR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAnPoC,SAmPzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApPhB,OAsPO,W;O;KAlQX,C;oGAeA,yB;MAAA,+D;MAsPA,gD;MAtPA,uC;QAAY,kBAAiB,gB;QAqPR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAtPoC,SAsPzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAvPhB,OAyPO,W;O;KArQX,C;oGAeA,yB;MAAA,+D;MAyPA,gD;MAzPA,uC;QAAY,kBAAiB,gB;QAwPR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAzPoC,SAyPzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1PhB,OA4PO,W;O;KAxQX,C;oGAeA,yB;MAAA,+D;MA4PA,gD;MA5PA,uC;QAAY,kBAAiB,gB;QA2PR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA5PoC,SA4PzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7PhB,OA+PO,W;O;KA3QX,C;oGAeA,yB;MAAA,+D;MA+PA,gD;MA/PA,uC;QAAY,kBAAiB,gB;QA8PR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA/PoC,SA+PzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhQhB,OAkQO,W;O;KA9QX,C;oGAeA,yB;MAAA,+D;MAkQA,gD;MAlQA,uC;QAAY,kBAAiB,gB;QAiQR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAlQoC,SAkQzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAnQhB,OAqQO,W;O;KAjRX,C;oGAeA,yB;MAAA,+D;MAqQA,gD;MArQA,uC;QAAY,kBAAiB,gB;QAoQR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WArQoC,SAqQzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtQhB,OAwQO,W;O;KApRX,C;oGAeA,yB;MAAA,+D;MAwQA,gD;MAxQA,uC;QAAY,kBAAiB,gB;QAuQR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAxQoC,SAwQzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAzQhB,OA2QO,W;O;KAvRX,C;oGAeA,yB;MAAA,+D;MA2QA,gD;MA3QA,uC;QAAY,kBAAiB,gB;QA0QR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA3QoC,SA2QzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5QhB,OA8QO,W;O;KA1RX,C;oGAeA,yB;MAAA,+D;MA8QA,gD;MA9QA,uC;QAAY,kBAAiB,gB;QA6QR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA9QoC,SA8QzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/QhB,OAiRO,W;O;KA7RX,C;oGAeA,yB;MAAA,+D;MAiRA,gD;MAjRA,uC;QAAY,kBAAiB,gB;QAgRR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAjRoC,SAiRzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlRhB,OAoRO,W;O;KAhSX,C;oGAeA,yB;MAAA,+D;MAoRA,gD;MApRA,uC;QAAY,kBAAiB,gB;QAmRR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WApRoC,SAoRzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QArRhB,OAuRO,W;O;KAnSX,C;oGAeA,yB;MAAA,+D;MAuRA,gD;MAvRA,uC;QAAY,kBAAiB,gB;QAsRR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAvRoC,SAuRzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxRhB,OA0RO,W;O;KAtSX,C;oGAeA,yB;MAAA,+D;MA0RA,gD;MA1RA,uC;QAAY,kBAAiB,gB;QAyRR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA1RoC,SA0RzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA3RhB,OA6RO,W;O;KAzSX,C;oGAeA,yB;MAAA,+D;MA6RA,gD;MA7RA,uC;QAAY,kBAAiB,gB;QA4RR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA7RoC,SA6RzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9RhB,OAgSO,W;O;KA5TX,C;oGAeA,yB;MAAA,+D;MAgSA,gD;MAhSA,uC;QAAY,kBAAiB,gB;QA+RR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAhSoC,SAgSzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAjShB,OAmSO,W;O;KA/TX,C;oGAeA,yB;MAAA,+D;MAmSA,gD;MAnSA,uC;QAAY,kBAAiB,gB;QAkSR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAnSoC,SAmSzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApShB,OAsSO,W;O;KAlUX,C;oGAeA,yB;MAAA,+D;MAsSA,gD;MAtSA,uC;QAAY,kBAAiB,gB;QAqSR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAtSoC,SAsSzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAvShB,OAySO,W;O;KArUX,C;oGAeA,yB;MAAA,+D;MAySA,gD;MAzSA,uC;QAAY,kBAAiB,gB;QAwSR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAzSoC,SAySzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1ShB,OA4SO,W;O;KAxUX,C;oGAeA,yB;MAAA,+D;MA4SA,gD;MA5SA,uC;QAAY,kBAAiB,gB;QA2SR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA5SoC,SA4SzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7ShB,OA+SO,W;O;KA3UX,C;oGAeA,yB;MAAA,+D;MA+SA,gD;MA/SA,uC;QAAY,kBAAiB,gB;QA8SR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA/SoC,SA+SzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhThB,OAkTO,W;O;KA9UX,C;oGAeA,yB;MAAA,+D;MAkTA,gD;MAlTA,uC;QAAY,kBAAiB,gB;QAiTR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAlToC,SAkTzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAnThB,OAqTO,W;O;KAjVX,C;oGAeA,yB;MAAA,+D;MAqTA,gD;MArTA,uC;QAAY,kBAAiB,gB;QAoTR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WArToC,SAqTzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtThB,OAwTO,W;O;KApVX,C;oGAeA,yB;MAAA,+D;MAwTA,gD;MAxTA,uC;QAAY,kBAAiB,gB;QAuTR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAxToC,SAwTzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAzThB,OA2TO,W;O;KAvVX,C;oGAeA,yB;MAAA,+D;MA2TA,gD;MA3TA,uC;QAAY,kBAAiB,gB;QA0TR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA3ToC,SA2TzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5ThB,OA8TO,W;O;KA1VX,C;oGAeA,yB;MAAA,+D;MA8TA,gD;MA9TA,uC;QAAY,kBAAiB,gB;QA6TR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA9ToC,SA8TzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/ThB,OAiUO,W;O;KA7VX,C;oGAeA,yB;MAAA,+D;MAiUA,gD;MAjUA,uC;QAAY,kBAAiB,gB;QAgUR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAjUoC,SAiUzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlUhB,OAoUO,W;O;KAhWX,C;oGAeA,yB;MAAA,+D;MAoUA,gD;MApUA,uC;QAAY,kBAAiB,gB;QAmUR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WApUoC,SAoUzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QArUhB,OAuUO,W;O;KAnWX,C;oGAeA,yB;MAAA,+D;MAuUA,gD;MAvUA,uC;QAAY,kBAAiB,gB;QAsUR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAvUoC,SAuUzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxUhB,OA0UO,W;O;KAtWX,C;oGAeA,yB;MAAA,+D;MA0UA,gD;MA1UA,uC;QAAY,kBAAiB,gB;QAyUR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA1UoC,SA0UzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA3UhB,OA6UO,W;O;KAzWX,C;oGAeA,yB;MAAA,+D;MA6UA,gD;MA7UA,uC;QAAY,kBAAiB,gB;QA4UR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA7UoC,SA6UzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9UhB,OAgVO,W;O;KA5WX,C;oGAeA,yB;MAAA,+D;MAgVA,gD;MAhVA,uC;QAAY,kBAAiB,gB;QA+UR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAhVoC,SAgVzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAjVhB,OAmVO,W;O;KA/WX,C;oGAeA,yB;MAAA,+D;MAmVA,gD;MAnVA,uC;QAAY,kBAAiB,gB;QAkVR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAnVoC,SAmVzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApVhB,OAsVO,W;O;KAlXX,C;oGAeA,yB;MAAA,+D;MAsVA,gD;MAtVA,uC;QAAY,kBAAiB,gB;QAqVR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAtVoC,SAsVzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAvVhB,OAyVO,W;O;KArXX,C;oGAeA,yB;MAAA,+D;MAyVA,gD;MAzVA,uC;QAAY,kBAAiB,gB;QAwVR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAzVoC,SAyVzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1VhB,OA4VO,W;O;KAxXX,C;oGAeA,yB;MAAA,+D;MA4VA,gD;MA5VA,uC;QAAY,kBAAiB,gB;QA2VR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA5VoC,SA4VzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7VhB,OA+VO,W;O;KA3XX,C;oGAeA,yB;MAAA,+D;MA+VA,gD;MA/VA,uC;QAAY,kBAAiB,gB;QA8VR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA/VoC,SA+VzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhWhB,OAkWO,W;O;KA9XX,C;oGAeA,yB;MAAA,+D;MAkWA,gD;MAlWA,uC;QAAY,kBAAiB,gB;QAiWR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAlWoC,SAkWzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAnWhB,OAqWO,W;O;KAjYX,C;oGAeA,yB;MAAA,+D;MAqWA,gD;MArWA,uC;QAAY,kBAAiB,gB;QAoWR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WArWoC,SAqWzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtWhB,OAwWO,W;O;KApYX,C;oGAeA,yB;MAAA,+D;MAwWA,gD;MAxWA,uC;QAAY,kBAAiB,gB;QAuWR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAxWoC,SAwWzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAzWhB,OA2WO,W;O;KAvYX,C;oGAeA,yB;MAAA,+D;MA2WA,gD;MA3WA,uC;QAAY,kBAAiB,gB;QA0WR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA3WoC,SA2WzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5WhB,OA8WO,W;O;KA1YX,C;oGAeA,yB;MAAA,+D;MA8WA,gD;MA9WA,uC;QAAY,kBAAiB,gB;QA6WR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA9WoC,SA8WzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/WhB,OAiXO,W;O;KA7YX,C;oGAeA,yB;MAAA,+D;MAiXA,gD;MAjXA,uC;QAAY,kBAAiB,gB;QAgXR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAjXoC,SAiXzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlXhB,OAoXO,W;O;KAhZX,C;oGAeA,yB;MAAA,+D;MAoXA,gD;MApXA,uC;QAAY,kBAAiB,gB;QAmXR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WApXoC,SAoXzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QArXhB,OAuXO,W;O;KAnZX,C;oGAeA,yB;MAAA,+D;MAuXA,gD;MAvXA,uC;QAAY,kBAAiB,gB;QAsXR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAvXoC,SAuXzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxXhB,OA0XO,W;O;KAtZX,C;oGAeA,yB;MAAA,+D;MA0XA,gD;MA1XA,uC;QAAY,kBAAiB,gB;QAyXR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA1XoC,SA0XzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA3XhB,OA6XO,W;O;KAzZX,C;oGAeA,yB;MAAA,+D;MA6XA,gD;MA7XA,uC;QAAY,kBAAiB,gB;QA4XR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WA7XoC,SA6XzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9XhB,OAgYO,W;O;KA5ZX,C;oGAeA,yB;MAAA,+D;MAgYA,gD;MAhYA,uC;QAAY,kBAAiB,gB;QA+XR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAhYoC,SAgYzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAjYhB,OAmYO,W;O;KA/ZX,C;oGAeA,yB;MAAA,+D;MAmYA,gD;MAnYA,uC;QAAY,kBAAiB,gB;QAkYR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAnYoC,SAmYzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApYhB,OAsYO,W;O;KAlaX,C;oGAeA,yB;MAAA,+D;MAsYA,gD;MAtYA,uC;QAAY,kBAAiB,gB;QAqYR,gB;QADhB,YAAY,C;QACZ,iD;UAAgB,cAAhB,e;UACI,WAtYoC,SAsYzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY

EAAO,IAAP,C;;QAEhB,OAAO,W;O;KAfX,C;uFAkBA,yB;MAAA,gD;MAAA,oD;QAIoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,WAAW,UAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAEhB,OAAO,W;O;KARX,C;0FAW

J,C;;QAvQT,OAyQO,W;O;KAlRX,C;sFAYA,yB;MAAA,wE;MAyQA,oC;MAAA,+D;MAAA,gC;MAzQA,yC;QASW,kBAAU,oB;QAyQD,Q;QAAhB,iD;UAAgB,cAAhB,0B;UACI,UA1QoD,WA0Q1C,CAAY,oBAAZ,C;UO5hVP,U;UADP,YP8hVe,WO9hVH,WP8hVwB,GO9hVxB,C;UACL,IAAI,aAAJ,C;YACH,aP4hVuC,gB;YAA5B,WO3hVX,aP2hVgC,GO3hVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPwhVA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QA5QT,OA8QO,W;O;KAvRX,C;sFAYA,yB;MAAA,wE;MA8QA,+D;MA9QA,yD;QAUW,kBAAU,oB;QA8QD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA/QiD,WA+QvC,CAAY,OAAZ,C;UO9iVP,U;UADP,YPgjVe,WOhjVH,WPgjVwB,GOhjVxB,C;UACL,IAAI,aAAJ,C;YACH,aP8iVuC,gB;YAA5B,WO7iVX,aP6iVgC,GO7iVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP0iVA,iB;UACA,IAAK,WAjRyD,cAiRrD,CAAe,OAAf,C;;QAjRT,OAmRO,W;O;KA7RX,C;sFAaA,yB;MAAA,wE;MAmRA,+D;MAnRA,yD;QAUW,kBAAU,oB;QAmRD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UApRiD,WAoRvC,CAAY,OAAZ,C;UOhkVP,U;UADP,YPkkVe,WOlkVH,WPkkVwB,GOlkVxB,C;UACL,IAAI,aAAJ,C;YACH,aPgkVuC,gB;YAA5B,WO/jVX,aP+jVgC,GO/jVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP4jVA,iB;UACA,IAAK,WAtRyD,cAsRrD,CAAe,OAAf,C;;QAtRT,OAwRO,W;O;KAlSX,C;uFAaA,yB;MAAA,wE;MAwRA,+D;MAxRA,yD;QAUW,kBAAU,oB;QAwRD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAzRiD,WAyRvC,CAAY,OAAZ,C;UOllVP,U;UADP,YPolVe,WOplVH,WPolVwB,GOplVxB,C;UACL,IAAI,aAAJ,C;YACH,aPklVuC,gB;YAA5B,WOjlVX,aPilVgC,GOjlVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP8kVA,iB;UACA,IAAK,WA3RyD,cA2RrD,CAAe,OAAf,C;;QA3RT,OA6RO,W;O;KAvSX,C;uFAaA,yB;MAAA,wE;MA6RA,+D;MA7RA,yD;QAUW,kBAAU,oB;QA6RD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA9RiD,WA8RvC,CAAY,OAAZ,C;UOnmVP,U;UADP,YPqmVe,WOrmVH,WPqmVwB,GOrmVxB,C;UACL,IAAI,aAAJ,C;YACH,aPmmVuC,gB;YAA5B,WOlmVX,aPkmVgC,GOlmVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP+lVA,iB;UACA,IAAK,WAhSyD,cAgSrD,CAAe,OAAf,C;;QAhST,OAkSO,W;O;KA5SX,C;uFAaA,yB;MAAA,wE;MAkSA,+D;MAlSA,yD;QAUW,kBAAU,oB;QAkSD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAnSiD,WAmSvC,CAAY,OAAZ,C;UOpnVP,U;UADP,YPsnVe,WOtnVH,WPsnVwB,GOtnVxB,C;UACL,IAAI,aAAJ,C;YACH,aPonVuC,gB;YAA5B,WOnnVX,aPmnVgC,GOnnVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPgnVA,iB;UACA,IAAK,WArSyD,cAqSrD,CAAe,OAAf,C;;QArST,OAuSO,W;O;KAjTX,C;uFAaA,yB;MAAA,wE;MAuSA,+D;MAvSA,yD;QAUW,kBAAU,oB;QAuSD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAxSiD,WAwSvC,CAAY,OAAZ,C;UOroVP,U;UADP,YPuoVe,WOvoVH,WPuoVwB,GOvoVxB,C;UACL,IAAI,aAAJ,C;YACH,aPqoVuC,gB;YAA5B,WOpoVX,aPooVgC,GOpoVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPioVA,iB;UACA,IAAK,WA1SyD,cA0SrD,CAAe,OAAf,C;;QA1ST,OA4SO,W;O;KAtTX,C;uFAaA,yB;MAAA,wE;MA4SA,+D;MA5SA,yD;QAUW,kBAAU,oB;QA4SD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA7SiD,WA6SvC,CAAY,OAAZ,C;UOtpVP,U;UADP,YPwpVe,WOxpVH,WPwpVwB,GOxpVxB,C;UACL,IAAI,aAAJ,C;YACH,aPspVuC,gB;YAA5B,WOrpVX,aPqpVgC,GOrpVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPkpVA,iB;UACA,IAAK,WA/SyD,cA+SrD,CAAe,OAAf,C;;QA/ST,OAiTO,W;O;KA3TX,C;uFAaA,yB;MAAA,wE;MAiTA,+D;MAjTA,yD;QAUW,kBAAU,oB;QAiTD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAlTiD,WAkTvC,CAAY,OAAZ,C;UOvqVP,U;UADP,YPyqVe,WOzqVH,WPyqVwB,GOzqVxB,C;UACL,IAAI,aAAJ,C;YACH,aPuqVuC,gB;YAA5B,WOtqVX,aPsqVgC,GOtqVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPmqVA,iB;UACA,IAAK,WApTyD,cAoTrD,CAAe,OAAf,C;;QApTT,OAsTO,W;O;KAhUX,C;uFAaA,yB;MAAA,wE;MAsTA,+D;MAtTA,yD;QAUW,kBAAU,oB;QAsTD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAvTiD,WAuTvC,CAAY,OAAZ,C;UOxrVP,U;UADP,YP0rVe,WO1rVH,WP0rVwB,GO1rVxB,C;UACL,IAAI,aAAJ,C;YACH,aPwrVuC,gB;YAA5B,WOvrVX,aPurVgC,GOvrVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPorVA,iB;UACA,IAAK,WAzTyD,cAyTrD,CAAe,OAAf,C;;QAzTT,OA2TO,W;O;KArUX,C;uFAaA,yB;MAAA,wE;MA2TA,+D;MA3TA,yD;QAUW,kBAAU,oB;QA2TD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA5TiD,WA4TvC,CAAY,OAAZ,C;UOzsVP,U;UADP,YP2sVe,WO3sVH,WP2sVwB,GO3sVxB,C;UACL,IAAI,aAAJ,C;YACH,aPysVuC,gB;YAA5B,WOxsVX,aPwsVgC,GOxsVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPqsVA,iB;UACA,IAAK,WA9TyD,cA8TrD,CAAe,OAAf,C;;QA9TT,OAgUO,W;O;KA1UX,C;uFAaA,yB;MAAA,wE;MAgUA,+D;MAhUA,yD;QAUW,kBAAU,oB;QAgUD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAjUiD,WAiUvC,CAAY,OAAZ,C;UO1tVP,U;UADP,YP4tVe,WO5tVH,WP4tVwB,GO5tVxB,C;UACL,IAAI,aAAJ,C;YACH,aP0tVuC,gB;YAA5B,WOztVX,aPytVgC,GOztVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPstVA,iB;UACA,IAAK,WAnUyD,cAmUrD,CAAe,OAAf,C;;QAnUT,OAqUO,W;O;KA/UX,C;uFAaA,yB;MAAA,wE;MAqUA,+D;MArUA,yD;QAUW,kBAAU,oB;QAqUD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAtUiD,WAsUvC,CAAY,OAAZ,C;UO3uVP,U;UADP,YP6uVe,WO7uVH,WP6uVwB,GO7uVxB,C;UACL,IAAI,aAAJ,C;YACH,aP2uVuC,gB;YAA5B,WO1uVX,aP0uVgC,GO1uVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPuuVA,iB;UACA,IAAK,WAxUyD,cAwUrD,CAAe,OAAf,C;;QAxUT,OA0UO,W;O;KApVX,C;uFAaA,yB;MAAA,wE;MA0UA,+D;MA1UA,yD;QAUW,kBAAU,oB;QA0UD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA3UiD,WA2UvC,CAAY,OAAZ,C;UO5vVP,U;UADP,YP8vVe,WO9vVH,WP8vVwB,GO9vVxB,C;UACL,IAAI,aAAJ,C;YACH,aP4vVuC,gB;YAA5B,WO3vVX,aP2vVgC,GO3vVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPwvVA,iB;UACA,IAAK,WA7UyD,cA6UrD,CAAe,OAAf,C;;QA7UT,OA+UO,W;O;KAzVX,C;uFAaA,yB;MAAA,wE;MA+UA,+D;MA/UA,yD;QAUW,kBAAU,oB;QA+UD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAhViD,WAgVvC,CAAY,OAAZ,C;UO7wVP,U;UADP,YP+wVe,WO/wVH,WP+wVwB,GO/wVxB,C;UACL,IAAI,aAAJ,C;YACH,aP6wVuC,gB;YAA5B,WO5wVX,aP4wVgC,GO5wVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPywVA,iB;UACA,IAAK,WAlVyD,cAkVrD,CAAe,OAAf,C;;QAlVT,OAoVO,W;O;KA9VX,C;uFAaA,yB;MAAA,wE;MAoVA,+D;MApVA,yD;QAUW,kBAAU,oB;QAoVD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UArViD,WAqVvC,CAAY,OAAZ,C;UO9xVP,U;UADP,YPgyVe,WOhyVH,WPgyVwB,GOhyVxB,C;UACL,IAAI,aAAJ,C;YACH,aP8xVuC,gB;YAA5B,WO7xVX,aP6xVgC,GO7xVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP0xVA,iB;UACA,IAAK,WAvVyD,cAuVrD,CAAe,OAAf,C;;QAvVT,OAyVO,W;O;KAnWX,C;uFAaA,yB;MAAA,wE;MAyVA,+D;MAzVA,yD;QAUW,kBAAU,oB;QAyVD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA1ViD,WA0VvC,CAAY,OAAZ,C;UO/yVP,U;UADP,YPizVe,WOjzVH,WPizVwB,GOjzVxB,C;UACL,IAAI,aAAJ,C;YACH,aP+yVuC,gB;YAA5B,WO9yVX,aP8yVgC,GO9yVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP2yVA,iB;UACA,IAAK,WA5VyD,cA4VrD,CAAe,OAAf,C;;QA5VT,OA8VO,W;O;KAxWX,C;uFAaA,yB;MAAA,wE;MA8VA,+D;MA9VA,yD;QAUW,kBAAU,oB;QA8VD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA/ViD,WA+VvC,CAAY,OAAZ,C;UOh0VP,U;UADP,YPk0Ve,WOl0VH,WPk0VwB,GOl0VxB,C;UACL,IAAI,aAAJ,C;YACH,aPg0VuC,gB;YAA5B,WO/zVX,aP+zVgC,GO/zVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP4zVA,iB;UACA,IAAK,WAjWyD,cAiWrD,CAAe,OAAf,C;;QAjWT,OAmWO,W;O;KA7WX,C;uFAaA,yB;MAAA,wE;MAmWA,+D;MAnWA,yD;QAUW,kBAAU,oB;QAmWD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UApWiD,WAoWvC,CAAY,OAAZ,C;UOj1VP,U;UADP,YPm1Ve,WOn1VH,WPm1VwB,GOn1VxB,C;UACL,IAAI,aAAJ,C;YACH,aPi1VuC,gB;YAA5B,WOh1VX,aPg1VgC,GOh1VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP60VA,iB;UACA,IAAK,WAtWyD,cAsWrD,CAAe,OAAf,C;;QAtWT,OAwWO,W;O;KAlXX,C;uFAaA,yB;MAAA,wE;MAwWA,+D;MAxWA,yD;QAUW,kBAAU,oB;QAwWD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAzWiD,WAyWvC,CAAY,OAAZ,C;UOl2VP,U;UADP,YPo2Ve,WOp2VH,WPo2VwB,GOp2VxB,C;UACL,IAAI,aAAJ,C;YACH,aPk2VuC,gB;YAA5B,WOj2VX,aPi2VgC,GOj2VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP82VA,iB;UACA,IAAK,WA3WyD,cA2WrD,CAAe,OAAf,C;;QA3WT,OA6WO,W;O;KAvXX,C;uFAaA,yB;MAAA,wE;MA6WA,+D;MA7WA,yD;QAUW,kBAAU,oB;QA6WD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA9WiD,WA8WvC,CAAY,OAAZ,C;UOn3VP,U;UADP,YPq3Ve,WOr3VH,WPq3VwB,GOr3VxB,C;UACL,IAAI,aAAJ,C;YACH,aPm3VuC,gB;YAA5B,WOl3VX,aPk3VgC,GOl3VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP+2VA,iB;UACA,IAAK,WAhXyD,cAgXrD,CAAe,OAAf,C;;QAhXT,OAkXO,W;O;KA5XX,C;uFAaA,yB;MAAA,wE;MAkXA,+D;MAlXA,yD;QAUW,kBAAU,oB;QAkXD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAnXiD,WAmXvC,CAAY,OAAZ,C;UOp4VP,U;UADP,YPs4Ve,WOt4VH,WPs4VwB,GOt4VxB,C;UACL,IAAI,aAAJ,C;YACH,aPo4VuC,gB;YAA5B,WOn4VX,aPm4VgC,GOn4VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPg4VA,iB;UACA,IAAK,WArXyD,cAqXrD,CAAe,OAAf,C;;QArXT,OAuXO,W;O;KAjYX,C;uFAaA,yB;MAAA,wE;MAuXA,+D;MAvXA,yD;QAUW,kBAAU,oB;QAuXD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAxXiD,WAwXvC,CAAY,OAAZ,C;UOr5VP,U;UADP,YPu5Ve,WOv5VH,WPu5VwB,GOv5VxB,C;UACL,IAAI,aAAJ,C;YACH,aPq5VuC,gB;YAA5B,WOp5VX,aPo5VgC,GOp5VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPi5VA,iB;UACA,IAAK,WA1XyD,cA0XrD,CAAe,OAAf,C;;QA1XT,OA4XO,W;O;KAtYX,C;uFAaA,yB;MAAA,wE;MA4XA,+D;MA5XA,yD;QAUW,kBAAU,oB;QA4XD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA7XiD,WA6XvC,CAAY,OAAZ,C;UOt6VP,U;UADP,YPw6Ve,WOx6VH,WPw6VwB,GOx6VxB,C;UACL,IAAI,aAAJ,C;YACH,aPs6VuC,gB;YAA5B,WOr6VX,aPq6VgC,GOr6VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPk6VA,iB;UACA,IAAK,WA/XyD,cA+XrD,CAAe,OAAf,C;;QA/XT,OAiYO,W;O;KA3YX,C;uFAaA,yB;MAAA,wE;MAiYA,+D;MAjYA,yD;QAUW,kBAAU,oB;QAiYD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAlYiD,WAkYvC,CAAY,OAAZ,C;UOv7VP,U;UADP,YPy7Ve,WOz7VH,WPy7VwB,GOz7VxB,C;UACL,IAAI,aAAJ,C;YACH,aPu7VuC,gB;YAA5B,WOt7VX,aPs7VgC,GOt7VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPm7VA,iB;UACA,IAAK,WApYyD,cAoYrD,CAAe,OAAf,C;;QApYT,OAsYO,W;O;KAhZX,C;uFAaA,yB;MAAA,wE;MAsYA,+D;MAtYA,yD;QAUW,kBAAU,oB;QAsYD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAvYiD,WAuYvC,CAAY,OAAZ,C;UOx8VP,U;UADP,YP08Ve,WO18VH,WP08VwB,GO18VxB,C;UACL,IAAI,aAAJ,C;YACH,aPw8VuC,gB;YAA5B,WOv8VX,aPu8VgC,GOv8VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPo8VA,iB;UACA,IAAK,WAzYyD,cAyYrD,CAAe,OAAf,C;;QAzYT,OA2YO,W;O;KArZX,C;uFAaA,yB;MAAA,wE;MA2YA,+D;MA3YA,yD;QAUW,kBAAU,oB;QA2YD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA5YiD,WA4YvC,CAAY,OAAZ,C;UOz9VP,U;UADP,YP29Ve,WO39VH,WP29VwB,GO39VxB,C;UACL,IAAI,aAAJ,C;YACH,aPy9VuC,gB;YAA5B,WOx9VX,aPw9VgC,GOx9VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPq9VA,iB;UACA,IAAK,WA9YyD,cA8YrD,CAAe,OAAf,C;;QA9YT,OAgZO,W;O;KA1ZX,C;uFAaA,yB;MAAA,wE;MAgZA,+D;MAhZA,yD;QAUW,kBAAU,oB;QAgZD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAjZiD,WAiZvC,CAAY,OAAZ,C;UO1+VP,U;UADP,YP4+Ve,WO5+VH,WP4+VwB,GO5+VxB,C;UACL,IAAI,aAAJ,C;YACH,aP0+VuC,gB;YAA5B,WOz+VX,aPy+VgC,GOz+VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPs+VA,iB;UACA,IAAK,WAnZyD,cAmZrD,CAAe,OAAf,C;;QAnZT,OAqZO,W;O;KA/ZX,C;uFAaA,yB;MAAA,wE;MAqZA,+D;MArZA,yD;QAUW,kBAAU,oB;QAqZD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAtZiD,WAsZvC,CAAY,OAAZ,C;UO3/VP,U;UADP,YP6/Ve,WO7/VH,WP6/VwB,GO7/VxB,C;UACL,IAAI,aAAJ,C;YACH,aP2/VuC,gB;YAA5B,WO1/VX,aP0/VgC,GO1/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPu/VA,iB;UACA,IAAK,WAxZyD,cAwZrD,CAAe,OAAf,C;;QAxZT,OA0ZO,W;O;KApaX,C;uFAaA,yB;MAAA,wE;MA0ZA,+D;MA1ZA,yD;QAUW,kBAAU,oB;QA0ZD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA3ZiD,WA2ZvC,CAAY,OAAZ,C;UO5gWP,U;UADP,YP8gWe,WO9gWH,WP8gWwB,GO9gWxB,C;UACL,IAAI,aAAJ,C;YACH,aP4gWuC,gB;YAA5B,WO3gWX,aP2gWgC,GO3gWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPwgWA,iB;UACA,IAAK,WA7ZyD,cA6ZrD,CAAe,OAAf,C;;QA7ZT,OA+ZO,W;O;KAzaX,C;uFAaA,yB;MAAA,wE;MA+ZA,+D;MA/ZA,yD;QAUW,kBAAU,oB;QA+ZD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAhaiD,WAgavC,CAAY,OAAZ,C;UO7hWP,U;UADP,YP+hWe,WO/hWH,WP+hWwB,GO/hWxB,C;UACL,IAAI,aAAJ,C;YACH,aP6hWuC,gB;YAA5B,WO5hWX,aP4hWgC,GO5hWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPyhWA,iB;UACA,IAAK,WAlayD,cAkarD,CAAe,OAAf,C;;QAlaT,OAoaO,W;O;KA9aX,C;uFAaA,yB;MAAA,wE;MAoaA,+D;MApaA,yD;QAUW,kBAAU,oB;QAoaD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAraiD,WAqavC,CAAY,OAAZ,C;UO9iWP,U;UADP,YPgjWe,WOhjWH,WPgjWwB,GOhjWxB,C;UACL,IAAI,aAAJ,C;YACH,aP8iWuC,gB;YAA5B,WO7iWX,aP6iWgC,GO7iWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP0iWA,iB;UACA,IAAK,WAvayD,cAuarD,CAAe,OAAf,C;;QAvaT,OAyaO,W;O;KAnbX,C;uFAaA,yB;MAAA,wE;MAyaA,+D;MAzaA,yD;QAUW,kBAAU,oB;QAyaD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA1aiD,WA0avC,CAAY,OAAZ,C;UO/jWP,U;UADP,YPikWe,WOjkWH,WPikWwB,GOjkWxB,C;UACL,IAAI,aAAJ,C;YACH,aP+jWuC,gB;YAA5B,WO9jWX,aP8jWgC,GO9jWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP2jWA,iB;UACA,IAAK,WA5byD,cA4brD,CAAe,OAAf,C;;QA5bT,OA8bO,W;O;KAxcX,C;uFAaA,yB;MAAA,wE;MA8bA,+D;MA9bA,yD;QAUW,kBAAU,oB;QA8bD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA/biD,WA+bvC,CAAY,OAAZ,C;UOhlWP,U;UADP,YPklWe,WOllWH,WPklWwB,GOllWxB,C;UACL,IAAI,aAAJ,C;YACH,aPglWuC,gB;YAA5B,WO/kWX,aP+kWgC,GO/kWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP4kWA,iB;UACA,IAAK,WAjcyD,cAicrD,CAAe,OAAf,C;;QAjcT,OAmcO,W;O;KA7cX,C;uFAaA,yB;MAAA,wE;MAmcA,+D;MAncA,yD;QAUW,kBAAU,oB;QAmcD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UApciD,WAocvC,CAAY,OAAZ,C;UOjmWP,U;UADP,YPmmWe,WOnmWH,WPmmWwB,GOnmWxB,C;UACL,IAAI,aAAJ,C;YACH,aPimWuC,gB;YAA5B,WOhmWX,aPgmWgC,GOhmWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP6lWA,iB;UACA,IAAK,WAtcyD,cAscrD,CAAe,OAAf,C;;QAtcT,OAwcO,W;O;KAldX,C;uFAaA,yB;MAAA,wE;MAwcA,+D;MAxcA,yD;QAUW,kBAAU,oB;QAwcD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAzciD,WAycvC,CAAY,OAAZ,C;UOlnWP,U;UADP,YPonWe,WOpnWH,WPonWwB,GOpnWxB,C;UACL,IAAI,aAAJ,C;YACH,aPknWuC,gB;YAA5B,WOjnWX,aPinWgC,GOjnWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP8mWA,iB;UACA,IAAK,WA3cyD,cA2crD,CAAe,OAAf,C;;QA3cT,OA6cO,W;O;KAveX,C;uFAaA,yB;MAAA,wE;MA6cA,+D;MA7cA,yD;QAUW,kBAAU,oB;QA6cD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA9ciD,WA8cvC,CAAY,OAAZ,C;UOnoWP,U;UADP,YPqoWe,WOroWH,WPqoWwB,GOroWxB,C;UACL,IAAI,aAAJ,C;YACH,aPmoWuC,gB;YAA5B,WOloWX,aPkoWgC,GOloWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP+nWA,iB;UACA,IAAK,WAhdyD,cAgdrD,CAAe,OAAf,C;;QAhdT,OAkdO,W;O;KA5dX,C;uFAaA,yB;MAAA,wE;MAkdA,+D;MAldA,yD;QAUW,kBAAU,oB;QAkdD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAndiD,WAmdvC,CAAY,OAAZ,C;UOppWP,U;UADP,YPspWe,WOtpWH,WPspWwB,GOtpWxB,C;UACL,IAAI,aAAJ,C;YACH,aPopWuC,gB;YAA5B,WOnpWX,aPmpWgC,GOnpWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPgpWA,iB;UACA,IAAK,WArdyD,cAqdrD,CAAe,OAAf,C;;QArdT,OAudO,W;O;KAjeX,C;uFAaA,yB;MAAA,wE;MAudA,+D;MAvdA,yD;QAUW,kBAAU,oB;QAudD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAxdiD,WAwdvC,CAAY,OAAZ,C;UOrqWP,U;UADP,YPuqWe,WOvqWH,WPuqWwB,GOvqWxB,C;UACL,IAAI,aAAJ,C;YACH,aPqqWuC,gB;YAA5B,WOpqWX,aPoqWgC,GOpqWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPiqWA,iB;UACA,IAAK,WA1dyD,cA0drD,CAAe,OAAf,C;;QA1dT,OA4dO,W;O;KAtePX,C;uFAaA,yB;MAAA,wE;MA4dA,+D;MA5dA,yD;QAUW,kBAAU,oB;QA4dD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA7diD,WA6dvC,CAAY,OAAZ,C;UOtrWP,U;UADP,YPwrWe,WOxrWH,WPwrWwB,GOxrWxB,C;UACL,IAAI,aAAJ,C;YACH,aPsrWuC,gB;YAA5B,WOrrWX,aPqrWgC,GOrrWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPkrWA,iB;UACA,IAAK,WA/dyD,cA+drD,CAAe,OAAf,C;;QA/dT,OAieO,W;O;KAzeX,C;uFAaA,yB;MAAA,wE;MAieA,+D;MAjeA,yD;QAUW,kBAAU,oB;QAieD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAleiD,WAkevC,CAAY,OAAZ,C;UOvsWP,U;UADP,YPysWe,WOzsWH,WPysWwB,GOzsWxB,C;UACL,IAAI,aAAJ,C;YACH,aPusWuC,gB;YAA5B,WOtsWX,aPssWgC,GOtsWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPmsWA,iB;UACA,IAAK,WApeyD,cAoerD,CAAe,OAAf,C;;QApeT,OAseO,W;O;KA9eX,C;uFAaA,yB;MAAA,wE;MAseA,+D;MAteA,yD;QAUW,kBAAU,oB;QAseD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAveiD,WAuevC,CAAY,OAAZ,C;UOxtWP,U;UADP,YP0tWe,WO1tWH,WP0tWwB,GO1tWxB,C;UACL,IAAI,aAAJ,C;YACH,aPwtWuC,gB;YAA5B,WOvtWX,aPutWgC,GOvtWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPotWA,iB;UACA,IAAK,WAzeyD,cAyerD,CAAe,OAAf,C;;QAzeT,OA2eO,W;O;KAnfX,C;uFAaA,yB;MAAA,wE;MA2eA,+D;MA3eA,yD;QAUW,kBAAU,oB;QA2eD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA5eiD,WA4evC,CAAY,OAAZ,C;UOzuWP,U;UADP,YP2uWe,WO3uWH,WP2uWwB,GO3uWxB,C;UACL,IAAI,aAAJ,C;YACH,aPyuWuC,gB;YAA5B,WOxuWX,aPwuWgC,GOxuWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPquWA,iB;UACA,IAAK,WA9eyD,cA8erD,CAAe,OAAf,C;;QA9eT,OAgfO,W;O;KAxfX,C;uFAaA,yB;MAAA,wE;MAgfA,+D;MAhfA,yD;QAUW,kBAAU,oB;QAgfD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAjfiD,WAifvC,CAAY,OAAZ,C;UO1vWP,U;UADP,YP4vWe,WO5vWH,WP4vWwB,GO5vWxB,C;UACL,IAAI,aAAJ,C;YACH,aP0vWuC,gB;YAA5B,WOzvWX,aPyvWgC,GOzvWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPsvWA,iB;UACA,IAAK,WAnfyD,cAmfrD,CAAe,OAAf,C;;QAnfT,OAqfO,W;O;KA7fX,C;uFAaA,yB;MAAA,wE;MAqfA,+D;MArfA,yD;QAUW,kBAAU,oB;QAqfD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAtfiD,WAsfvC,CAAY,OAAZ,C;UO3wWP,U;UADP,YP6wWe,WO7wWH,WP6wWwB,GO7wWxB,C;UACL,IAAI,aAAJ,C;YACH,aP2wWuC,gB;YAA5B,WO1wWX,aP0wWgC,GO1wWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPuwWA,iB;UACA,IAAK,WAxfyD,cAwfrD,CAAe,OAAf,C;;QAxfT,OA0fO,W;O;KAlgBX,C;uFAaA,yB;MAAA,wE;MA0fA,+D;MA1fA,yD;QAUW,kBAAU,oB;QA0fD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA3fiD,WA2fvC,CAAY,OAAZ,C;UO5xWP,U;UADP,YP8xWe,WO9xWH,WP8xWwB,GO9xWxB,C;UACL,IAAI,aAAJ,C;YACH,aP4xWuC,gB;YAA5B,WO3xWX,aP2xWgC,GO3xWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPwxWA,iB;UACA,IAAK,WA7fyD,cA6frD,CAAe,OAAf,C;;QA7fT,OA+fO,W;O;KAvgBX,C;uFAaA,yB;MAAA,wE;MA+fA,+D;MA/fA,yD;QAUW,kBAAU,oB;QA+fD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAhgBiD,WAgfvC,CAAY,OAAZ,C;UO7yWP,U;UADP,YP+yWe,WO/yWH,WP+yWwB,GO/yWxB,C;UACL,IAAI,aAAJ,C;YACH,aP6yWuC,gB;YAA5B,WO5yWX,aP4yWgC,GO5yWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPyyWA,iB;UACA,IAAK,WAlgByD,cAkgBrD,CAAe,OAAf,C;;QAlgBT,OAogBO,W;O;KA5gBX,C;uFAaA,yB;MAAA,wE;MAogBA,+D;MApgBA,yD;QAUW,kBAAU,oB;QAogBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UArgBiD,WAqgBvC,CAAY,OAAZ,C;UO9zWP,U;UADP,YPg0We,WOh0WH,WPg0WwB,GOh0WxB,C;UACL,IAAI,aAAJ,C;YACH,aP8zWuC,gB;YAA5B,WO7zWX,aP6zWgC,GO7zWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP0zWA,iB;UACA,IAAK,WAvgByD,cAugBrD,CAAe,OAAf,C;;QAvgBT,OAygBO,W;O;KAjhBX,C;uFAaA,yB;MAAA,wE;MAygBA,+D;MAzgBA,yD;QAUW,kBAAU,oB;QAygBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA1gBiD,WA0gBvC,CAAY,OAAZ,C;UO/0WP,U;UADP,YPi1We,WOj1WH,WPi1WwB,GOj1WxB,C;UACL,IAAI,aAAJ,C;YACH,aP+0WuC,gB;YAA5B,WO90WX,aP80WgC,GO90WhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP20WA,iB;UACA,IAAK,WA5gByD,cA4gBrD,CAAe,OAAf,C;;QA5gBT,OA8gBO,W;O;KAthBX,C;uFAaA,yB;MAAA,wE;MA8gBA,+D;MA9gBA,yD;QAUW,kBAAU,oB;QA8gBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA/gBiD,WA+gBvC,CAAY,OAAZ,C;UOh2WP,U;UADP,YPk2We,WOl2WH,WPk2WwB,GOl2WxB,C;UACL,IAAI,aAAJ,C;YACH,aPg2WuC,gB;YAA5B,WO/1WX,aP+1WgC,GO/1WhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP41WA,iB;UACA,IAAK,WAjhByD,cAihBrD,CAAe,OAAf,C;;QAjhBT,OAmhBO,W;O;KA3hBX,C;uFAaA,yB;MAAA,wE;MAmhBA,+D;MAnhBA,yD;QAUW,kBAAU,oB;QAmhBD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAphBiD,WAohBvC,CAAY,OAAZ,C;UOj3WP,U;UADP,YPm3We,WOn3WH,WPm3WwB,GOn3WxB,C;UACL,IAAI,aAAJ,C;YACH,aPi3WuC,gB;YAA5B,WOh3WX,aPg3WgC,GOh3WhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP63WA,iB;UACA,IAAK,WAthByD,cAshBrD,CAAe,OAAf,C;;QAthB

UxB,C;UACL,IAAI,aAAJ,C;YACH,aPs7UuC,gB;YAA5B,WOr7UX,aPq7UgC,GOr7UhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPk7UA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOv8UP,U;UADP,YPy8Ue,WOz8UH,WPy8UwB,GOz8UxB,C;UACL,IAAI,aAAJ,C;YACH,aPu8UuC,gB;YAA5B,WOt8UX,aPs8UgC,GOt8UhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPm8UA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOx9UP,U;UADP,YP09Ue,WO19UH,WP09UwB,GO19UxB,C;UACL,IAAI,aAAJ,C;YACH,aPw9UuC,gB;YAA5B,WOv9UX,aPu9UgC,GOv9UhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPo9UA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOz+UP,U;UADP,YP2+Ue,WO3+UH,WP2+UwB,GO3+UxB,C;UACL,IAAI,aAAJ,C;YACH,aPy+UuC,gB;YAA5B,WOx+UX,aPw+UgC,GOx+UhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPq+UA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO1/UP,U;UADP,YP4/Ue,WO5/UH,WP4/UwB,GO5/UxB,C;UACL,IAAI,aAAJ,C;YACH,aP0/UuC,gB;YAA5B,WOz/UX,aPy/UgC,GOz/UhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPs/UA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO3gVP,U;UADP,YP6gVe,WO7gVH,WP6gVwB,GO7gVxB,C;UACL,IAAI,aAAJ,C;YACH,aP2gVuC,gB;YAA5B,WO1gVX,aP0gVgC,GO1gVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPugVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO5hVP,U;UADP,YP8hVe,WO9hVH,WP8hVwB,GO9hVxB,C;UACL,IAAI,aAAJ,C;YACH,aP4hVuC,gB;YAA5B,WO3hVX,aP2hVgC,GO3hVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPwhVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO9iVP,U;UADP,YPgjVe,WOhjVH,WPgjVwB,GOhjVxB,C;UACL,IAAI,aAAJ,C;YACH,aP8iVuC,gB;YAA5B,WO7iVX,aP6iVgC,GO7iVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP0iVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOjkVP,U;UADP,YPmkVe,WOnkVH,WPmkVwB,GOnkVxB,C;UACL,IAAI,aAAJ,C;YACH,aPikVuC,gB;YAA5B,WOhkVX,aPgkVgC,GOhkVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP6jVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOplVP,U;UADP,YPslVe,WOtlVH,WPslVwB,GOtlVxB,C;UACL,IAAI,aAAJ,C;YACH,aPolVuC,gB;YAA5B,WOnlVX,aPmlVgC,GOnlVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPglVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOvmVP,U;UADP,YPymVe,WOzmVH,WPymVwB,GOzmVxB,C;UACL,IAAI,aAAJ,C;YACH,aPumVuC,gB;YAA5B,WOtmVX,aPsmVgC,GOtmVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPmmVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO1nVP,U;UADP,YP4nVe,WO5nVH,WP4nVwB,GO5nVxB,C;UACL,IAAI,aAAJ,C;YACH,aP0nVuC,gB;YAA5B,WOznVX,aPynVgC,GOznVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPsnVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO7oVP,U;UADP,YP+oVe,WO/oVH,WP+oVwB,GO/oVxB,C;UACL,IAAI,aAAJ,C;YACH,aP6oVuC,gB;YAA5B,WO5oVX,aP4oVgC,GO5oVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPyoVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOhqVP,U;UADP,YPkqVe,WOlqVH,WPkqVwB,GOlqVxB,C;UACL,IAAI,aAAJ,C;YACH,aPgqVuC,gB;YAA5B,WO/pVX,aP+pVgC,GO/pVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP4pVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOnrVP,U;UADP,YPqrVe,WOrrVH,WPqrVwB,GOrrVxB,C;UACL,IAAI,aAAJ,C;YACH,aPmrVuC,gB;YAA5B,WOlrVX,aPkrVgC,GOlrVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP+qVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOtsVP,U;UADP,YPwsVe,WOxsVH,WPwsVwB,GOxsVxB,C;UACL,IAAI,aAAJ,C;YACH,aPssVuC,gB;YAA5B,WOrsVX,aPqsVgC,GOrsVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPksVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOztVP,U;UADP,YP2tVe,WO3tVH,WP2tVwB,GO3tVxB,C;UACL,IAAI,aAAJ,C;YACH,aPytVuC,gB;YAA5B,WOxtVX,aPwtVgC,GOxtVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPqtVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO5uVP,U;UADP,YP8uVe,WO9uVH,WP8uVwB,GO9uVxB,C;UACL,IAAI,aAAJ,C;YACH,aP4uVuC,gB;YAA5B,WO3uVX,aP2uVgC,GO3uVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPwuVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO/vVP,U;UADP,YPiwVe,WOjwVH,WPiwVwB,GOjwVxB,C;UACL,IAAI,aAAJ,C;YACH,aP+vVuC,gB;YAA5B,WO9vVX,aP8vVgC,GO9vVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP2vVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOlxVP,U;UADP,YPoxVe,WOpxVH,WPoxVwB,GOpxVxB,C;UACL,IAAI,aAAJ,C;YACH,aPkxVuC,gB;YAA5B,WOjxVX,aPixVgC,GOjxVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP8wVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOryVP,U;UADP,YPuyVe,WOvyVH,WPuyVwB,GOvyVxB,C;UACL,IAAI,aAAJ,C;YACH,aPqyVuC,gB;YAA5B,WOpyVX,aPoyVgC,GOpyVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPiyVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOxzVP,U;UADP,YP0zVe,WO1zVH,WP0zVwB,GO1zVxB,C;UACL,IAAI,aAAJ,C;YACH,aPwzVuC,gB;YAA5B,WOvzVX,aPuzVgC,GOvzVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPozVA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO30VP,U;UADP,YP60Ve,WO70VH,WP60VwB,GO70VxB,C;UACL,IAAI,aAAJ,C;YACH,aP20VuC,gB;YAA5B,WO10VX,aP00VgC,GO10VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPu0VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO51VP,U;UADP,YP81Ve,WO91VH,WP81VwB,GO91VxB,C;UACL,IAAI,aAAJ,C;YACH,aP41VuC,gB;YAA5B,WO31VX,aP21VgC,GO31VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPw1VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO72VP,U;UADP,YP+2Ve,WO/2VH,WP+2VwB,GO/2VxB,C;UACL,IAAI,aAAJ,C;YACH,aP62VuC,gB;YAA5B,WO52VX,aP42VgC,GO52VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPy2VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO93VP,U;UADP,YPg4Ve,WOh4VH,WPg4VwB,GOh4VxB,C;UACL,IAAI,aAAJ,C;YACH,aP83VuC,gB;YAA5B,WO73VX,aP63VgC,GO73VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP03VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO/4VP,U;UADP,YPi5Ve,WOj5VH,WPi5VwB,GOj5VxB,C;UACL,IAAI,aAAJ,C;YACH,aP+4VuC,gB;YAA5B,WO94VX,aP84VgC,GO94VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP24VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOh6VP,U;UADP,YPk6Ve,WOl6VH,WPk6VwB,GOl6VxB,C;UACL,IAAI,aAAJ,C;YACH,aPg6VuC,gB;YAA5B,WO/5VX,aP+5VgC,GO/5VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP45VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOj7VP,U;UADP,YPm7Ve,WOn7VH,WPm7VwB,GOn7VxB,C;UACL,IAAI,aAAJ,C;YACH,aPi7VuC,gB;YAA5B,WOh7VX,aPg7VgC,GOh7VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP67VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOl8VP,U;UADP,YPo8Ve,WOp8VH,WPo8VwB,GOp8VxB,C;UACL,IAAI,aAAJ,C;YACH,aPk8VuC,gB;YAA5B,WOj8VX,aPi8VgC,GOj8VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP88VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOn9VP,U;UADP,YPq9Ve,WOr9VH,WPq9VwB,GOr9VxB,C;UACL,IAAI,aAAJ,C;YACH,aPm9VuC,gB;YAA5B,WOl9VX,aPk9VgC,GOl9VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP+9VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOp+VP,U;UADP,YPs+Ve,WOt+VH,WPs+VwB,GOt+VxB,C;UACL,IAAI,aAAJ,C;YACH,aPo+VuC,gB;YAA5B,WOn+VX,aPm+VgC,GOn+VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPg+VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOr/VP,U;UADP,YPu/Ve,WOv/VH,WPu/VwB,GOv/VxB,C;UACL,IAAI,aAAJ,C;YACH,aPq/VuC,gB;YAA5B,WOp/VX,aPo/VgC,GOp/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPi/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOt/VP,U;UADP,YPw/Ve,WOx/VH,WPw/VwB,GOx/VxB,C;UACL,IAAI,aAAJ,C;YACH,aPs/VuC,gB;YAA5B,WOr/VX,aPq/VgC,GOr/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPk/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOv/VP,U;UADP,YPy/Ve,WOz/VH,WPy/VwB,GOz/VxB,C;UACL,IAAI,aAAJ,C;YACH,aPu/VuC,gB;YAA5B,WOt/VX,aPs/VgC,GOt/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPm/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOx/VP,U;UADP,YP0/Ve,WO1/VH,WP0/VwB,GO1/VxB,C;UACL,IAAI,aAAJ,C;YACH,aPw/VuC,gB;YAA5B,WOv/VX,aPu/VgC,GOv/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPo/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOy/VP,U;UADP,YP2/Ve,WO3/VH,WP2/VwB,GO3/VxB,C;UACL,IAAI,aAAJ,C;YACH,aPy/VuC,gB;YAA5B,WOx/VX,aPw/VgC,GOx/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPq/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO1/VP,U;UADP,YP4/Ve,WO5/VH,WP4/VwB,GO5/VxB,C;UACL,IAAI,aAAJ,C;YACH,aP0/VuC,gB;YAA5B,WOz/VX,aPy/VgC,GOz/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPs/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO3/VP,U;UADP,YP6/Ve,WO7/VH,WP6/VwB,GO7/VxB,C;UACL,IAAI,aAAJ,C;YACH,aP2/VuC,gB;YAA5B,WO1/VX,aP0/VgC,GO1/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPu/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO5/VP,U;UADP,YP8/Ve,WO9/VH,WP8/VwB,GO9/VxB,C;UACL,IAAI,aAAJ,C;YACH,aP4/VuC,gB;YAA5B,WO3/VX,aP2/VgC,GO3/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPw/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO7/VP,U;UADP,YP+/Ve,WO//VH,WP+/VwB,GO//VxB,C;UACL,IAAI,aAAJ,C;YACH,aP6/VuC,gB;YAA5B,WO5/VX,aP4/VgC,GO5/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPy/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO9/VP,U;UADP,YPggWe,WOhgWH,WPggWwB,GOhgWxB,C;UACL,IAAI,aAAJ,C;YACH,aP8/VuC,gB;YAA5B,WO7/VX,aP6/VgC,GO7/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP0/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO//VP,U;UADP,YPigWe,WOjgWH,WPigWwB,GOjgWxB,C;UACL,IAAI,aAAJ,C;YACH,aP+/VuC,gB;YAA5B,WO9/VX,aP8/VgC,GO9/VhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP2/VA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOhhWP,U;UADP,YPkhWe,WOlhWH,WPkhWwB,GOlhWxB,C;UACL,IAAI,aAAJ,C;YACH,aPghWuC,gB;YAA5B,WO/gWX,aP+gWgC,GO/gWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP4gWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOjhWP,U;UADP,YPmhWe,WOnhWH,WPmhWwB,GOnhWxB,C;UACL,IAAI,aAAJ,C;YACH,aPihWuC,gB;YAA5B,WOhhWX,aPghWgC,GOhhWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP6gWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOlhWP,U;UADP,YPohWe,WOphWH,WPohWwB,GOphWxB,C;UACL,IAAI,aAAJ,C;YACH,aPkhWuC,gB;YAA5B,WOjhWX,aPihWgC,GOjhWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP8gWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOnhWP,U;UADP,YPqhWe,WOrhWH,WPqhWwB,GOrhWxB,C;UACL,IAAI,aAAJ,C;YACH,aPmhWuC,gB;YAA5B,WOlhWX,aPkhWgC,GOlhWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UP+gWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOphWP,U;UADP,YPshWe,WOthWH,WPshWwB,GOthWxB,C;UACL,IAAI,aAAJ,C;YACH,aPohWuC,gB;YAA5B,WOnhWX,aPmhWgC,GOnhWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPghWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOrhWP,U;UADP,YPuhWe,WOvhWH,WPuhWwB,GOvhWxB,C;UACL,IAAI,aAAJ,C;YACH,aPqhWuC,gB;YAA5B,WOphWX,aPohWgC,GOphWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPihWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOthWP,U;UADP,YPwhWe,WOxhWH,WPwhWwB,GOxhWxB,C;UACL,IAAI,aAAJ,C;YACH,aPshWuC,gB;YAA5B,WOrhWX,aPqhWgC,GOrhWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPkhWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOvhWP,U;UADP,YPyhWe,WOzhWH,WPyhWwB,GOzhWxB,C;UACL,IAAI,aAAJ,C;YACH,aPuhWuC,gB;YAA5B,WOthWX,aPshWgC,GOthWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPmhWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOxhWP,U;UADP,YP0hWe,WO1hWH,WP0hWwB,GO1hWxB,C;UACL,IAAI,aAAJ,C;YACH,aPwhWuC,gB;YAA5B,WOvhWX,aPuhWgC,GOvhWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPohWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOzhWP,U;UADP,YP2hWe,WO3hWH,WP2hWwB,GO3hWxB,C;UACL,IAAI,aAAJ,C;YACH,aPyhWuC,gB;YAA5B,WOxhWX,aPwhWgC,GOxhWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPqhWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO1hWP,U;UADP,YP4hWe,WO5hWH,WP4hWwB,GO5hWxB,C;UACL,IAAI,aAAJ,C;YACH,aP0hWuC,gB;YAA5B,WOzhWX,aPyhWgC,GOzhWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPshWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO3hWP,U;UADP,YP6hWe,WO7hWH,WP6hWwB,GO7hWxB,C;UACL,IAAI,aAAJ,C;YACH,aP2hWuC,gB;YAA5B,WO1hWX,aP0hWgC,GO1hWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPuhWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO5hWP,U;UADP,YP8hWe,WO9hWH,WP8hWwB,GO9hWxB,C;UACL,IAAI,aAAJ,C;YACH,aP4hWuC,gB;YAA5B,WO3hWX,aP2hWgC,GO3hWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPwhWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO7hWP,U;UADP,YP+hWe,WO/hWH,WP+hWwB,GO/hWxB,C;UACL,IAAI,aAAJ,C;YACH,aP6hWuC,gB;YAA5B,WO5hWX,aP4hWgC,GO5hWhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPyhWA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAgB,

MAAA,+D;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UO5qVP,U;UADP,YP8qVe,WO9qVH,WP8qVwB,GO9qVxB,C;UACL,IAAI,aAAJ,C;YACH,aP4qVuC,gB;YAA5B,WO3qVX,aP2qVgC,GO3qVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPwqVA,iB;UACA,IAAK,WAAI,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;O;KAfX,C;2FAkBA,yB;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UACI,UAAU,YAAY,oBAAZ,C;UO9rVP,U;UADP,YPgsVe,WOhsVH,WPgsVwB,GOhsVxB,C;UACL,IAAI,aAAJ,C;YACH,aP8rVuC,gB;YAA5B,WO7rVX,aP6rVgC,GO7rVhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UPurVA,iB;UACA,IAAK,WAAI,eAAe,oBAAf,CAAJ,C;;QAET,OAAO,W;O;KAfX,C;0FAkBA,yB;MAAA,kC;MAAA,4C;MAAA,wE;QAAQ,kBAAM,eAAa,gBAAb,C;QAAd,+B;QAAA,iD;UAAA,WAAc

Db,YAAY,C;QACZ,iD;UAAa,WAAb,e;UApiEmC,U;UAAA,yBAoiET,cApiES,EAoiET,sBApiES,WAoiEA,IApiEA,W;YAA6C,6B;;;QAChF,OAAO,W;O;KARX,C;8FAWA,6C;MAQiB,UACiB,M;MAF9B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAAY,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;MAChB,OAAO,W;K;gGAGX

X,wBAAU,SAAV,gB;UAAU,QAAA,SAAV,M;UACI,UAAU,SAAS,CAAT,C;UACV,IAAI,GAAI,WAAI,GAAJ,CAAR,C;YACI,IAAK,WAAI,CAAJ,C;;QAEb,OAAO,I;O;KAhBX,C;4FAmBA,yB;MAAA,2D;MAAA,+D;MAAA,sC;QAWc,Q;QAFV,UAAU,c;QACV,WAAW,gB;QACX

e,eAAL,SAAK,C;MACX,OAAJ,GAAI,EAAO,KAAP,C;MACJ,OAAO,G;K;IAGX,mC;MAUI,UAAe,eAAL,SAA
K,C;MACX,OAAJ,GAAI,EAAO,KAAP,C;MACJ,OAAO,G;K;4EAGX,gC;MAMoB,Q;MAAhB,wBAAgB,SAAhB
,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I
;K;8EAGX,gC;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAA
U,OAAV,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I;K;8EAGX,gC;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB
;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I;K;
8EAGX,gC;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,O
AAV,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I;K;8EAGX,gC;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;Q
AAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I;K;8E
AGX,gC;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OA
AV,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I;K;8EAGX,gC;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QA
AgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I;K;8EA
GX,gC;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAA
V,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I;K;8EAGX,yB;MAAA,oC;MAAA,gC;MAAA,uC;QAMoB,Q;QA
AhB,wBAAgB,SAAhB,gB;QAAgB,cAAhB,UAAgB,SAAhB,O;UAAsB,IAAI,CAAC,UAAU,oBAAV,CAAL,C;Y
AAyB,OAAO,K;;QACtD,OAAO,I;O;KAPX,C;IAUA,wB;MAMI,OAAO,EA5mJA,qBAAQ,CA4mJR,C;K;IAGX,0
B;MAMI,OAAO,EA7mJA,qBAAQ,CA6mJR,C;K;IAGX,0B;MAMI,OAAO,EA9mJA,qBAAQ,CA8mJR,C;K;IAG
X,0B;MAMI,OAAO,EA/mJA,qBAAQ,CA+mJR,C;K;IAGX,0B;MAMI,OAAO,EAhnJA,qBAAQ,CAgnJR,C;K;IA
GX,0B;MAMI,OAAO,EAjnJA,qBAAQ,CAinJR,C;K;IAGX,0B;MAMI,OAAO,EAlnJA,qBAAQ,CAknJR,C;K;IAG
X,0B;MAMI,OAAO,EAnnJA,qBAAQ,CAmnJR,C;K;IAGX,0B;MAMI,OAAO,EApnJA,qBAAQ,CAonJR,C;K;8E
AGX,gC;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,I;;MACrD,O
AAO,K;K;8EAGX,gC;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,I;;MACrD,O
AAO,K;K;+

C,OAAO,K;K;mFAGX,yB;MAAA,oC;MAAA,gC;MAAA,uC;QAKoB,Q;QADhB,YAAY,C;QACZ,wBAAgB,SA
AhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UAAsB,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,qB;;QAC9C,OAAO
,K;O;KANX,C;8EASA,yC;MAUoB,Q;MADhB,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAh
B,M;QAAsB,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;K;gFAGX,yC;MAUoB,Q;MADhB,kB
AAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,cAAc,UAAU,WAAV,EAAuB,OAAv
B,C;;MACpC,OAAO,W;K;gFAGX,yC;MAUoB,Q;MADhB,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,c
AAA,SAAhB,M;QAAsB,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;K;gFAGX,yC;MAUoB,Q;
MADhB,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,cAAc,UAAU,WAAV,EA
AuB,OAAvB,C;;MACpC,OAAO,W;K;gFAGX,yC;MAUoB,Q;MADhB,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB
;QAAgB,cAAA,SAAhB,M;QAAsB,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;K;gFAGX,yC;
MAUoB,Q;MADhB,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,cAAc,UAAU,
WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;K;gFAGX,yC;MAUoB,Q;MADhB,kBAAkB,O;MAClB,wBAAgB
,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;K;g
FAGX,yC;MAUoB,Q;MADhB,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,cA
Ac,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;K;gFAGX,yB;MAAA,oC;MAAA,gC;MAAA,gD;QAU
oB,Q;QADhB,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UAAsB,cAAc,UAA
U,WAAV,EAAuB,oBAAvB,C;;QACpC,OAAO,W;O;KAXX,C;4FAcA,yC;MAYoB,UAA8B,M;MAF9C,YAAY,C
;MACZ,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,cAAc,WAAU,cAAV,EAA
U,sBAAV,WAAmB,WAAnB,EAAgC,OAAhC,C;;MACpC,OAAO,W;K;8FAGX,yC;MAYoB,UAA8B,M;MAF9C,
YAAY,C;MACZ,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,cAAc,WAAU,cA
AV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OAAhC,C;;MACpC,OAAO,W;K;8FAGX,yC;MAYoB,UAA8B,M
;MAF9C,YAAY,C;MACZ,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,cAAc,W
AAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OAAhC,C;;MACpC,OAAO,W;K;8FAGX,yC;MAYoB,U
AA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB
,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OAAhC,C;;MACpC,OAAO,W;K;8FAGX,yC;M
AYoB,UAA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M
;QAAsB,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OAAhC,C;;MACpC,OAAO,W;K;8FAG
X,yC;MAYoB,UAA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,S
AAhB,M;QAAsB,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OAAhC,C;;MACpC,OAAO,W
;K;8FAGX,yC;MAYoB,UAA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MAClB,wBAAgB,SAAhB,gB;QAAgB,
cAAA,SAAhB,M;QAAsB,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OAAhC,C;;MACpC,
OAAO,W;K;8FAGX,yC;MAYoB,UAA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MAClB,wBAAgB,SAAhB,g
B;QAAgB,cAAA,SAAhB,M;QAAsB,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OAAhC,C;
;MACpC,OAAO,W;K;8FAGX,yB;MAAA,oC;MAAA,gC;MAAA,gD;QAYoB,UAA8B,M;QAF9C,YAAY,C;QAC
Z,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UAAsB,cAAc,WAAU,cAAV,E
AAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QACpC,OAAO,W;O;KAbX,C;wFAgBA,yB;MAAA,8D;M
AAA,gD;QAYoC,Q;QAHhC,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU
,UAAI,YAAJ,EAAI,oBAAJ,OAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,8D;
MAAA,gD;QAYoC,Q;QAHhC,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UA
AU,UAAI,YAAJ,EAAI,oBAAJ,OAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,8
D;MAAA,gD;QAYoC,Q;QAHhC,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,U
AAU,UAAI,YAAJ,EAAI,oBAAJ,OAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,
8D;MAAA,gD;QAYoC,Q;QAHhC,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,
UAAU,UAAI,YAAJ,EAAI,oBAAJ,OAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAdX,C;0FAiBA,yB;MAA
A,8D;MAAA,gD;QAYoC,Q;QAHhC,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAA
c,UAAU,UAAI,YAAJ,EAAI,oBAAJ,OAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAdX,C;0FAiBA,yB;MA
AA,8D;MAAA,gD;QAYoC,Q;QAHhC,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cA
Ac,UAAU,UAAI,YAAJ,EAAI,oBAAJ,OAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAdX,C;0FAiBA,yB;M

AAA,8D;MAAA,gD;QAYoC,Q;QAHhC,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,YAAJ,EAAI,oBAAJ,OAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,8D;MAAA,gD;QAYoC,Q;QAHhC,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,YAAJ,EAAI,oBAAJ,OAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAdX,C;0FAiBA,yB;MAAA,8D;MAAA,oC;MAAA,gD;QAYoC,Q;QAHhC,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,sBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAdX,C;sGAiBA,yB;MAAA,8D;MAAA,gD;QAUI,Y;QAAJ,kBAAI,O;QAAJ,OAAI,W;O;KAVR,C;wGAa

O,sB;K;IAGX,0B;MAGI,OAAO,sB;K;IAGX,0B;MAGI,OAAO,sB;K;IAGX,0B;MAGI,OAAO,sB;K;IAGX,0B;MAGI,OAAO,sB;K;IAGX,0B;MAGI,OAAO,sB;K;IAGX,0B;MAGI,OAAO,sB;K;IAGX,0B;MAGI,OAAO,sB;K;gFAGX,yB;MAsDA,8D;MAtDA,sC;QAGW,sB;;UA0DP,IAhxLO,qBAAQ,CAgxLf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA9DmB,QA8DJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAjEe,QAiEP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAvEP,yB;O;KAHJ,C;kFAMA,yB;MAuEA,8D;MAvEA,sC;QAGW,sB;;UA2EP,IA/xLO,qBAAQ,CA+xLf,C+;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA/EmB,mB+EmB,QA/EnB,GA+EmB,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAlFe,QAkFP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAxFP,yB;O;KAHJ,C;kFAMA,yB;MAwFA,8D;MAxFA,sC;QAGW,sB;;UA4FP,IAhzLO,qBAAQ,CAgzLf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAhGmB,QAgGJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAnGe,QAmGP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAzGP,yB;O;KAHJ,C;kFAMA,yB;MAyGA,8D;MAzGA,sC;QAGW,sB;;UA6GP,IAj1LO,qBAAQ,CAi1Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAjHmB,QAiHJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QApHe,QAoHP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA1HP,yB;O;KAHJ,C;kFAMA,yB;MA0HA,8D;MA1HA,sC;QAGW,sB;;UA8HP,IAl2LO,qBAAQ,CAk2Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAlImB,QAkIJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QArIe,QAqIP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA3IP,yB;O;KAHJ,C;kFAMA,yB;MA2IA,8D;MA3IA,sC;QAGW,sB;;UA+IP,IAn3LO,qBAAQ,CAm3Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAnJmB,QAmJJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAtJe,QAsJP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA5JP,yB;O;KAHJ,C;kFAMA,yB;MA4JA,8D;MA5JA,sC;QAGW,sB;;UAgKP,IAp4LO,qBAAQ,CAo4Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eApKmB,QAoKJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAvKe,QAuKP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA7KP,yB;O;KAHJ,C;kFAMA,yB;MA6KA,8D;MA7KA,sC;QAGW,sB;;UAiLP,IAr5LO,qBAAQ,CAq5Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eArLmB,QAqLJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAxLe,QAwLP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA9LP,yB;O;KAHJ,C;kFAMA,yB;MA8LA,8D;MA9LA,sC;QAGW,sB;;UAkMP,IAt6LO,qBAAQ,CAs6Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAtMmB,QAsMJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAzMe,QAyMP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA/MP,yB;O;KAHJ,C;kFAMA,yB;MA+MA,8D;MA/MA,sC;QAGW,sB;;UAmNP,IAv7LO,qBAAQ,CAu7Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAvNmB,QAuNJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA1Ne,QA0NP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAhOP,yB;O;KAHJ,C;kFAMA,yB;MAgOA,8D;MAhOA,sC;QAGW,sB;;UAoOP,IAx8LO,qBAAQ,CAw8Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAxOmB,QAwOJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA3Oe,QA2OP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAjPP,yB;O;KAHJ,C;kFAMA,yB;MAiPA,8D;MAjPA,sC;QAGW,sB;;UAqPP,IAz9LO,qBAAQ,CAy9Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAzPmB,QAyPJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA5Pe,QA4PP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAlQP,yB;O;KAHJ,C;kFAMA,yB;MAkQA,8D;MAlQA,sC;QAGW,sB;;UAsQP,IA1+LO,qBAAQ,CA0+Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA1QmB,QA0QJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA7Qe,QA6QP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAnRP,yB;O;KAHJ,C;kFAMA,yB;MAmRA,8D;MAnRA,sC;QAGW,sB;;UAuRP,IA3/LO,qBAAQ,CA2/Lf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA3RmB,QA2RJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA9Re,QA8RP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QApSP,yB;O;KAHJ,C;kFAMA,yB;MAoSA,8D;MApSA,sC;QAGW,sB;;UAwSP,IA5gMO,qBAAQ,CA4gMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA5SmB,QA4SJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA/Se,QA+SP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QArTP,yB;O;KAHJ,C;kFAMA,yB;MAqTA,8D;MArTA,sC;QAGW,sB;;UAyTP,IA7hMO,qBAAQ,CA6hMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA7TmB,QA6TJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAhUe,QAgUP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAtUP,yB;O;KAHJ,C;kFAMA,yB;MAsUA,8D;MAtUA,sC;QAGW,sB;;UA0UP,IA9iMO,qBAAQ,CA8iMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA9UmB,QA8UJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAjVe,QAiVP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAvVP,yB;O;KAHJ,C;kFAMA,yB;MAuVA,8D;MAvVA,sC;QAGW,sB;;UA2VP,IA/kMO,qBAAQ,CA+kMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA/VmB,QA+VJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAlWe,QAkWP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAxWP,yB;O;KAHJ,C;kFAMA,yB;MAwWA,8D;MAxWA,sC;QAGW,sB;;UA4WP,IAhnMO,qBAAQ,CAgnMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAhXmB,QAgXJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAnXe,QAmXP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAzXP,yB;O;KAHJ,C;kFAMA,yB;MAyXA,8D;MAzXA,sC;QAGW,sB;;UA6XP,IAjpMO,qBAAQ,CAipMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAjYmB,QAiYJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QApYe,QAoYP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA1YP,yB;O;KAHJ,C;kFAMA,yB;MA0YA,8D;MA1YA,sC;QAGW,sB;;UA8YP,IAlrMO,qBAAQ,CAkrMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAlZmB,QAkZJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QArZe,QAqZP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA3ZP,yB;O;KAHJ,C;kFAMA,yB;MA2ZA,8D;MA3ZA,sC;QAGW,sB;;UA+ZP,IAnsMO,qBAAQ,CAmsMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAnanB,QAmaoB,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAtae,QAsaP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA5aP,yB;O;KAHJ,C;kFAMA,yB;MA4aA,8D;MA5aA,sC;QAGW,sB;;UAgbP,IAptMO,qBAAQ,CAotMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eApbmB,QAobJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAvbe,QAubP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA7bP,yB;O;KAHJ,C;kFAMA,yB;MA6bA,8D;MA7bA,sC;QAGW,sB;;UAicP,IArvMO,qBAAQ,CAqvMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eArcmB,QAqcJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAxce,QAwcP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA9cP,yB;O;KAHJ,C;kFAMA,yB;MA8cA,8D;MA9cA,sC;QAGW,sB;;UAkdP,IAtwMO,qBAAQ,CAswMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAtdmB,QAscJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAzde,QAydP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA/dP,yB;O;KAHJ,C;kFAMA,yB;MA+dA,8D;MA/dA,sC;QAGW,sB;;UAmeP,IAvxMO,qBAAQ,CAuxMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAveoB,QAudL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA1ee,QA0eP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAhfP,yB;O;KAHJ,C;kFAMA,yB;MAgfA,8D;MAhfA,sC;QAGW,sB;;UAofP,IAxyMO,qBAAQ,CAwyMf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAxfoB,QAwfL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA3ff,QA2fwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAjgBP,yB;O;KAHJ,C;kFAMA,yB;MAigBA,8D;MAjgBA,sC;QAGW,sB;;UAqgBP,IAz0MO,qBAAQ,CAy0Mf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAzgBoB,QAygBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA5gBf,QA4gBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAlhBP,yB;O;KAHJ,C;kFAMA,yB;MAkhBA,8D;MAlhBA,sC;QAGW,sB;;UAshBP,IA12MO,qBAAQ,CA02Mf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA1hBoB,QA0hBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA7hBf,QA6hBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAniBP,yB;O;KAHJ,C;kFAMA,yB;MAmiBA,8D;MAniBA,sC;QAGW,sB;;UAuiBP,IA34MO,qBAAQ,CA24Mf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA3iBoB,QA2iBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA9iBf,QA8iBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QApjBP,yB;O;KAHJ,C;kFAMA,yB;MAojBA,8D;MApjBA,sC;QAGW,sB;;UAwjBP,IA55MO,qBAAQ,CA45Mf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA5jBoB,QA4jBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA/jBf,QA+jBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QArkBP,yB;O;KAHJ,C;kFAMA,yB;MAqkBA,8D;MArkBA,sC;QAGW,sB;;UAykBP,IA76MO,qBAAQ,CA66Mf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA7kBoB,QA6kBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAhlBf,QAglBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAtlBP,yB;O;KAHJ,C;kFAMA,yB;MAslBA,8D;MAtlBA,sC;QAGW,sB;;UA0lBP,IA98MO,qBAAQ,CA88Mf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA9lBoB,QA8lBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAjmBf,QAimBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAvmBP,yB;O;KAHJ,C;kFAMA,yB;MAumBA,8D;MAvmBA,sC;QAGW,sB;;UA2mBP,IA/+MO,qBAAQ,CA++Mf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA/mBoB,QA+mBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAlnBf,QAknBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAxnBP,yB;O;KAHJ,C;kFAMA,yB;MAwnBA,8D;MAxnBA,sC;QAGW,sB;;UA4nBP,IAhhNO,qBAAQ,CAghNf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAhoBoB,QAgoBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAnoBf,QAmoBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAzoBP,yB;O;KAHJ,C;kFAMA,yB;MAyoBA,8D;MAzoBA,sC;QAGW,sB;;UA6oBP,IAjiNO,qBAAQ,CAiiNf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAjpBoB,QAipBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAppBf,QAopBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA1pBP,yB;O;KAHJ,C;kFAMA,yB;MA0pBA,8D;MA1pBA,sC;QAGW,sB;;UA8pBP,IAlkNO,qBAAQ,CAkkNf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAlqBoB,QAkqBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QArqBf,QAqqBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA3qBP,yB;O;KAHJ,C;kFAMA,yB;MA2qBA,8D;MA3qBA,sC;QAGW,sB;;UA+qBP,IAnlNO,qBAAQ,CAmlNf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAnrBoB,QAmrBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAtrBf,QAsrBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA5rBP,yB;O;KAHJ,C;kFAMA,yB;MA4rBA,8D;MA5rBA,sC;QAGW,sB;;UAgsBP,IAplNO,qBAAQ,CAolNf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eApsBoB,QAosBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAvsBf,QAusBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA7sBP,yB;O;KAHJ,C;kFAMA,yB;MA6sBA,8D;MA7sBA,sC;QAGW,sB;;UAitBP,IArmNO,qBAAQ,CAqmNf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eArtBoB,QAqtBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAxtBf,QAwtBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA9tBP,yB;O;KAHJ,C;kFAMA,yB;MA8tBA,8D;MA9tBA,sC;QAGW,sB;;UAkuBP,IAtnNO,qBAAQ,CAsnNf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAtuBoB,QAsuBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QAzuBf,QAyuBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA/uBP,yB;O;KAHJ,C;kFAMA,yB;MA+uBA,8D;MA/uBA,sC;QAGW,sB;;UAmvBP,IAvoNO,qBAAQ,CAuoNf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAvvBoB,QAuvBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA1vBf,QA0vBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAhwBP,yB;O;KAHJ,C;kFAMA,yB;MAgwBA,8D;MAhwBA,sC;QAGW,sB;;UAowBP,IAxpNO,qBAAQ,CAwpNf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAxwBoB,QAwwBL,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAL,C;YACR,QA3wBf,QA2wBwB,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAjxBP,yB;O;KAHJ,C;kFAMA,yB;MAixBA,8D;MAjxBA,sC;QAGW,sB;;UAqxBP,IAzqNO,qBAAQ,CAyqNf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBA

,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OA
AO,O;O;KApBX,C;8FAuBA,yB;MAAA,8D;MAAA,sC;QAOI,IA9yLO,qBAAQ,CA8yLf,C;UAAe,OAAO,I;QACt
B,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC
3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SA
AS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;O;KA
pBX,C;8FAuBA,yB;MAAA,8D;MAAA,sC;QAOI,IA7zLO,qBAAQ,CA6zLf,C;UAAe,OAAO,I;QACtB,cAAc,UA
AK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,S
AAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IA
AI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;O;KApBX,C;8FAuBA,yB;MAAA,8D;MAAA,sC;QAOI,IA50LO,qBAAQ,CA40Lf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;O;KApBX,C;8FAuBA,yB;MAAA,8D;MAAA,sC;QAOI,IA31LO,qBAAQ,CA21Lf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;O;KApBX,C;8FAuBA,yB;MAAA,8D;MAAA,sC;QAOI,IA12LO,qBAAQ,CA02Lf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;O;KApBX,C;8FAuBA,yB;MAAA,8D;MAAA,sC;QAOI,IAz3LO,qBAAQ,CAy3Lf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;O;KApBX,C;8FAuBA,yB;MAAA,8D;MAAA,sC;QAOI,IA12LO,qBAAQ,CA02

;QAeiB,Q;QAFb,IAzjMO,qBAAQ,CAyjMf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB7vbG,MAAO,KlB6vbO,QkB7vbP,ElB6vbiB,CkB7vbjB,C;;QlB+vbd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB1wbA,iB;MlB0wbA,sC;QAeiB,Q;QAFb,IAvkMO,qBAAQ,CAukMf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnxbG,MAAO,KlBmxbO,QkBnxbP,ElBmxbiB,CkBnxbjB,C;;QlBqxbd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,oC;MAAA,8D;MkBhybA,iB;MlBgybA,sC;QAeiB,Q;QAFb,IArlMO,qBAAQ,CAqlMf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WkBzzbG,MAAO,KlByzbO,QkBzzbP,ElByzbiB,CkBzzbjB,C;;QlB2zbd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBt0bA,iB;MlBs0bA,sC;QAeiB,Q;QAFb,IA3qMO,qBAAQ,CA2qMf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB10bG,MAAO,KlB00bO,QkB10bP,ElB00biB,CkB10bjB,C;;QlB40bd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBv1bA,iB;MlBu1bA,sC;QAeiB,Q;QAFb,IAzsMO,qBAAQ,CAysMf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB32bG,MAAO,KlB22bO,QkB32bP,ElB22biB,CkB32bjB,C;;QlB62bd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBx3bA,iB;MlBw3bA,sC;QAeiB,Q;QAFb,IAvuMO,qBAAQ,CAuuMf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB54bG,MAAO,KlB44bO,QkB54bP,ElB44biB,CkB54bjB,C;;QlB84bd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBz5bA,iB;MlBy5bA,sC;QAeiB,Q;QAFb,IAnwMO,qBAAQ,CAmwMf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB76bG,MAAO,KlB66bO,QkB76bP,ElB66biB,CkB76bjB,C;;QlB+6bd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB18bA,iB;MlB08bA,sC;QAeiB,Q;QAFb,IA/xMO,qBAAQ,CA+xMf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB99bG,MAAO,KlB89bO,QkB99bP,ElB89biB,CkB99bjB,C;;QlBg+bd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB3+bA,iB;MlB2+bA,sC;QAeiB,Q;QAFb,IAnzMO,qBAAQ,CAmzMf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/+bG,MAAO,KlB++bO,QkB/+bP,ElB++biB,CkB/+bjB,C;;QlBi/bd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB5/bA,iB;MlB4/bA,sC;QAeiB,Q;QAFb,IA/0MO,qBAAQ,CA+0Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBhhcG,MAAO,KlBghcO,QkBhhcP,ElBghciB,CkBhhcjB,C;;QlBkhcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB7hcA,iB;MlB6hcA,sC;QAeiB,Q;QAFb,IAn2MO,qBAAQ,CAm2Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBjjcG,MAAO,KlBijcO,QkBjjcP,ElBijciB,CkBjjcjB,C;;QlBmjcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB9jcA,iB;MlB8jcA,sC;QAeiB,Q;QAFb,IAv3MO,qBAAQ,CAu3Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBlkcG,MAAO,KlBkkcO,QkBlkcP,ElBkkciB,CkBlkcjB,C;;QlBokcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB/kcA,iB;MlB+kcA,sC;QAeiB,Q;QAFb,IA34MO,qBAAQ,CA24Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnlcG,MAAO,KlBmlcO,QkBnlcP,ElBmlciB,CkBnlcjB,C;;QlBqlcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBhmcA,iB;MlBgmcA,sC;QAeiB,Q;QAFb,IA/5MO,qBAAQ,CA+5Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBpmcG,MAAO,KlBomcO,QkBpmcP,ElBomciB,CkBpmcjB,C;;QlBsmcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBjncA,iB;MlBincA,sC;QAeiB,Q;QAFb,IAn8MO,qBAAQ,CAm8Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBrncG,MAAO,KlBqncO,QkBrncP,ElBqnciB,CkBrncjB,C;;QlBuncd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBlocA,iB;MlBkocA,sC;QAeiB,Q;QAFb,IAv9MO,qBAAQ,CAu9Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBtocG,MAAO,KlBsocO,QkBtocP,ElBsociB,CkBtocjB,C;;QlBwocd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBnpcA,iB;MlBmpcA,sC;QAeiB,Q;QAFb,IA3+MO,qBAAQ,CA2+Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvpcG,MAAO,KlBupcO,QkBvpcP,ElBupciB,CkBvpcjB,C;;QlBypcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBpqcA,iB;MlBoqcA,sC;QAeiB,Q;QAFb,IA/gNO,qBAAQ,CA+gNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBxqcG,MAAO,KlBwqcO,QkBxqcP,ElBwqciB,CkBxqcjB,C;;QlB0qcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBrrcA,iB;MlBqrcA,sC;QAeiB,Q;QAFb,IAnjNO,qBAAQ,CAmjNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBzrcG,MAAO,KlByrcO,QkBzrcP,ElByrciB,CkBzrcjB,C;;QlB2rcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBtscA,iB;MlBsscA,sC;QAeiB,Q;QAFb,IAvkNO,qBAAQ,CAukNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB1scG,MAAO,KlB0scO,QkB1scP,ElB0sciB,CkB1scjB,C;;QlB4scd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBvtcA,iB;MlButcA,sC;QAeiB,Q;QAFb,IA3lNO,qBAAQ,CA2lNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3tcG,MAAO,KlB2tcO,QkB3tcP,ElB2tciB,CkB3tcjB,C;;QlB6tcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBxucA,iB;MlBwucA,sC;QAeiB,Q;QAFb,IAnnNO,qBAAQ,CAmnNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB5ucG,MAAO,KlB4ucO,QkB5ucP,ElB4uciB,CkB5ucjB,C;;QlB8ucd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBzvcA,iB;MlByvcA,sC;QAeiB,Q;QAFb,IAvoNO,qBAAQ,CAuoNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB7vcG,MAAO,KlB6vcO,QkB7vcP,ElB6vciB,CkB7vcjB,C;;QlB+vcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB1wcA,iB;MlB0wcA,sC;QAeiB,Q;QAFb,IA3pNO,qBAAQ,CA2pNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB9wcG,MAAO,KlB8wcO,QkB9wcP,ElB8wciB,CkB9wcjB,C;;QlBgxcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB3xcA,iB;MlB2xcA,sC;QAeiB,Q;QAFb,IAnrNO,qBAAQ,CAmrNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/xcG,MAAO,KlB+xcO,QkB/xcP,ElB+xciB,CkB/xcjB,C;;QlBiycd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB5ycA,iB;MlB4ycA,sC;QAeiB,Q;QAFb,IAvsNO,qBAAQ,CAusNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBhzcG,MAAO,KlBgzcO,QkBhzcP,ElBgzciB,CkBhzcjB,C;;QlBkzcd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB70cA,iB;MlB60cA,sC;QAeiB,Q;QAFb,IA3tNO,qBAAQ,CA2tNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBj1cG,MAAO,KlBi1cO,QkBj1cP,ElBi1ciB,CkBj1cjB,C;;QlBm1cd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB92cA,iB;MlB82cA,sC;QAeiB,Q;QAFb,IAnvNO,qBAAQ,CAmvNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBl3cG,MAAO,KlBk3cO,QkBl3cP,ElBk3ciB,CkBl3cjB,C;;QlBo3cd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB/4cA,iB;MlB+4cA,sC;QAeiB,Q;QAFb,IAvwNO,qBAAQ,CAuwNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBn5cG,MAAO,KlBm5cO,QkBn5cP,ElBm5ciB,CkBn5cjB,C;;QlBq5cd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBh6cA,iB;MlBg6cA,sC;QAeiB,Q;QAFb,IA3xNO,qBAAQ,CA2xNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBp6cG,MAAO,KlBo6cO,QkBp6cP,ElBo6ciB,CkBp6cjB,C;;QlBs6cd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBj7cA,iB;MlBi7cA,sC;QAeiB,Q;QAFb,IAnyNO,qBAAQ,CAmyNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBr7cG,MAAO,KlBq7cO,QkBr7cP,ElBq7ciB,CkBr7cjB,C;;QlBu7cd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBl8cA,iB;MlBk8cA,sC;QAeiB,Q;QAFb,IAvzNO,qBAAQ,CAuzNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBt8cG,MAAO,KlBs8cO,QkBt8cP,ElBs8ciB,CkBt8cjB,C;;QlBw8cd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBn9cA,iB;MlBm9cA,sC;QAeiB,Q;QAFb,IA30NO,qBAAQ,CA20Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBv9cG,MAAO,KlBu9cO,QkBv9cP,ElBu9ciB,CkBv9cjB,C;;QlBy9cd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBp+cA,iB;MlBo+cA,sC;QAeiB,Q;QAFb,IA51NO,qBAAQ,CA41Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBx+cG,MAAO,KlBw+cO,QkBx+cP,ElBw+ciB,CkBx+cjB,C;;QlB0+cd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBr/cA,iB;MlBq/cA,sC;QAeiB,Q;QAFb,IA71NO,qBAAQ,CA61Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBz/cG,MAAO,KlBy/cO,QkBz/cP,ElBy/ciB,CkBz/cjB,C;;QlB2/cd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBtgdA,iB;MlBsgdA,sC;QAeiB,Q;QAFb,IA93NO,qBAAQ,CA83Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB1gdG,MAAO,KlB0gdO,QkB1gdP,ElB0gdiB,CkB1gdjB,C;;QlB4gdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBvhdA,iB;MlBuhdA,sC;QAeiB,Q;QAFb,IA/5NO,qBAAQ,CA+5Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3hdG,MAAO,KlB2hdO,QkB3hdP,ElB2hdiB,CkB3hdjB,C;;QlB6hdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBxidA,iB;MlBwidA,sC;QAeiB,Q;QAFb,IAh8NO,qBAAQ,CAg8Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB5idG,MAAO,KlB4idO,QkB5idP,ElB4idiB,CkB5idjB,C;;QlB8idd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBzjdA,iB;MlByjdA,sC;QAeiB,Q;QAFb,IAj+NO,qBAAQ,CAi+Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB7jdG,MAAO,KlB6jdO,QkB7jdP,ElB6jdiB,CkB7jdjB,C;;QlB+jdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB1kdA,iB;MlB0kdA,sC;QAeiB,Q;QAFb,IAl/NO,qBAAQ,CAk/Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB9kdG,MAAO,KlB8kdO,QkB9kdP,ElB8kdiB,CkB9kdjB,C;;QlBglddO,QAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB3ldA,iB;MlB2ldA,sC;QAeiB,Q;QAFb,IAngOO,qBAAQ,CAmgOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/ldG,MAAO,KlB+ldO,QkB/ldP,ElB+ldiB,CkB/ldjB,C;;QlBimdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB5mdA,iB;MlB4mdA,sC;QAeiB,Q;QAFb,IAphOO,qBAAQ,CAohOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBhndG,MAAO,KlBgndO,QkBhndP,ElBgndiB,CkBhndjB,C;;QlBkndd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB7ndA,iB;MlB6ndA,sC;QAeiB,Q;QAFb,IAriOO,qBAAQ,CAqiOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBjodG,MAAO,KlBiodO,QkBjodP,ElBiodiB,CkBjodjB,C;;QlBmodd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB9odA,iB;MlB8odA,sC;QAeiB,Q;QAFb,IAtjOO,qBAAQ,CAsjOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBlpdG,MAAO,KlBkpdO,QkBlpdP,ElBkpdiB,CkBlpdjB,C;;QlBopdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB/pdA,iB;MlB+pdA,sC;QAeiB,Q;QAFb,IAvkOO,qBAAQ,CAukOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnqdG,MAAO,KlBmqdO,QkBnqdP,ElBmqdiB,CkBnqdjB,C;;QlBqqdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBhrdA,iB;MlBgrdA,sC;QAeiB,Q;QAFb,IAxlOO,qBAAQ,CAwlOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBprdG,MAAO,KlBordO,QkBprdP,ElBordiB,CkBprdjB,C;;QlBsrdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBjsdA,iB;MlBisdA,sC;QAeiB,Q;QAFb,IAzmOO,qBAAQ,CAymOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBrsdG,MAAO,KlBqsdO,QkBrsdP,ElBqsdiB,CkBrsdjB,C;;QlBusdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBltdA,iB;MlBktdA,sC;QAeiB,Q;QAFb,IA1nOO,qBAAQ,CA0nOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBttdG,MAAO,KlBstdO,QkBttdP,ElBstdiB,CkBttdjB,C;;QlBwtdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBnudA,iB;MlBmudA,sC;QAeiB,Q;QAFb,IA3oOO,qBAAQ,CA2oOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvudG,MAAO,KlBuudO,QkBvudP,ElBuudiB,CkBvudjB,C;;QlBydd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBpvdA,iB;MlBovdA,sC;QAeiB,Q;QAFb,IA5pOO,qBAAQ,CA4pOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBxvdG,MAAO,KlBwvdO,QkBxvdP,ElBwvdiB,CkBxvdjB,C;;QlB0vdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBrwdA,iB;MlBqwdA,sC;QAeiB,Q;QAFb,IA7qOO,qBAAQ,CA6qOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBzwdG,MAAO,KlBywdO,QkBzwdP,ElBywdiB,CkBzwdjB,C;;QlB2wdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBtxdA,iB;MlBsxdA,sC;QAeiB,Q;QAFb,IA9rOO,qBAAQ,CA8rOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB1xdG,MAAO,KlB0xdO,QkB1xdP,ElB0xdiB,CkB1xdjB,C;;QlB4xdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBvydA,iB;MlBuydA,sC;QAeiB,Q;QAFb,IA/sOO,qBAAQ,CA+sOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3ydG,MAAO,KlB2ydO,QkB3ydP,ElB2yciB,CkB3ydjB,C;;QlB6ydd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBxzdA,iB;MlBwzdA,sC;QAeiB,Q;QAFb,IAhuOO,qBAAQ,CAguOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB5zdG,MAAO,KlB4zdO,QkB5zdP,ElB4zdiB,CkB5zdjB,C;;QlB8zdd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBz0dA,iB;MlBy0dA,sC;QAeiB,Q;QAFb,IAjvOO,qBAAQ,CAivOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB70dG,MAAO,KlB60dO,QkB70dP,ElB60diB,CkB70djB,C;;QlB+0dd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB12dA,iB;MlB02dA,sC;QAeiB,Q;QAFb,IAlwOO,qBAAQ,CAkwOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB92dG,MAAO,KlB82dO,QkB92dP,ElB82diB,CkB92djB,C;;QlBg3dd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB34dA,iB;MlB24dA,sC;QAeiB,Q;QAFb,IAnxOO,qBAAQ,CAmxOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/4dG,MAAO,KlB+4dO,QkB/4dP,ElB+4diB,CkB/4djB,C;;QlBi5dd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB56dA,iB;MlB46dA,sC;QAeiB,Q;QAFb,IApyOO,qBAAQ,CAoyOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBh7dG,MAAO,KlBg7dO,QkBh7dP,ElBg7diB,CkBh7djB,C;;QlBk7dd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB78dA,iB;MlB68dA,sC;QAeiB,Q;QAFb,IArzOO,qBAAQ,CAqzOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBj9dG,MAAO,KlBi9dO,QkBj9dP,ElBi9diB,CkBj9djB,C;;QlBm9dd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB9+dA,iB;MlB8+dA,sC;QAeiB,Q;QAFb,IAt0OO,qBAAQ,CAs0Of,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBl+dG,MAAO,KlBk+dO,QkBl+dP,ElBk+diB,CkBl+djB,C;;QlBo+dd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkB/+dA,iB;MlB++dA,sC;QAeiB,Q;QAFb,IA11OO,qBAAQ,CA01Of,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBn/dG,MAAO,KlB

QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAv6MO,qBAAQ,CAu6Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAr7MO,qBAAQ,CAq7Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn8MO,qBAAQ,CAm8Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAj9MO,qBAAQ,CAi9Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA/9MO,qBAAQ,CA+9Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA7+MO,qBAAQ,CA6+Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA3/MO,qBAAQ,CA2/Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAz/MO,qBAAQ,CAy/Mf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAvgNO,qBAAQ,CAugNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnhNO,qBAAQ,CAmhNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAjiNO,qBAAQ,CAiiNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA/iNO,qBAAQ,CA+iNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA7jNO,qBAAQ,CA6jNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA3kNO,qBAAQ,CA2kNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAzlNO,qBAAQ,CAylNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAvmNO,qBAAQ,CAumNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnnNO,qBAAQ,CAmnNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAjoNO,qBAAQ,CAioNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA/oNO,qBAAQ,CA+oNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA7pNO,qBAAQ,CA6pNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA3qNO,qBAAQ,CA2qNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAzrNO,qBAAQ,CAyrNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAvsNO,qBAAQ,CAusNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAntNO,qBAAQ,CAmtNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAjuNO,qBAAQ,CAiuNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA/uNO,qBAAQ,CA+uNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA7vNO,qBAAQ,CA6vNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA3wNO,qBAAQ,CA2wNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAzxNO,qBAAQ,CAyxNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAvyNO,qBAAQ,CAuyNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnzNO,qBAAQ,CAmzNf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAj0NO,qBAAQ,CAi0Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA/0NO,qBAAQ,CA+0Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA71NO,qBAAQ,CA61Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA32NO,qBAAQ,CA21Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAz3NO,qBAAQ,CAy3Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAv4NO,qBAAQ,CAu4Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn5NO,qBAAQ,CAm5Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAj6NO,qBAAQ,CAi6Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA/6NO,qBAAQ,CA+6Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA78NO,qBAAQ,CA68Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA38NO,qBAAQ,CA28Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAz9NO,qBAAQ,CAy9Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAv+NO,qBAAQ,CAu+Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn/NO,qBAAQ,CAm/Nf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAjgOO,qBAAQ,CAigOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA/gOO,qBAAQ,CA+gOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA7hOO,qBAAQ,CA6hOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA3iOO,qBAAQ,CA2iOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAzjOO,qBAAQ,CAyjOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAvkOO,qBAAQ,CAukOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnlOO,qBAAQ,CAmlOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAjmOO,qBAAQ,CAimOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA/mOO,qBAAQ,CA+mOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA7nOO,qBAAQ,CA6nOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA3oOO,qBAAQ,CA2oOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAzpOO,qBAAQ,CAypOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAvqOO,qBAAQ,CAuqOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnrOO,qBAAQ,CAmrOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAjsOO,qBAAQ,CAisOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA/sOO,qBAAQ,CA+sOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA7tOO,qBAAQ,CA6tOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA3uOO,qBAAQ,CA2uOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAzvOO,qBAAQ,CAyvOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAvwOO,qBAAQ,CAuwOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnxOO,qBAAQ,CAmxOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAjyOO,qBAAQ,CAiyOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,C

AU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB55cG,MAAO,KlB45cO,QkB55cP,ElB45c
iB,CkB55cjB,C;;QlB85cd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBz6cA,iB;MlBy6cA,sC;QAaiB,Q;Q
AFb,IAjwNO,qBAAQ,CAiwNf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,
aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBh7cG,MAAO,KlBg7cO,QkBh7cP,ElB
g7ciB,CkBh7cjB,C;;QlBk7cd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB77cA,iB;MlB67cA,sC;QAaiB,
Q;QAFb,IA7wNO,qBAAQ,CA6wNf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;Q
AAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBp8cG,MAAO,KlBo8cO,QkBp8c
P,ElBo8ciB,CkBp8cjB,C;;QlBs8cd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBj9cA,iB;MlBi9cA,sC;QA
aiB,Q;QAFb,IAzxNO,qBAAQ,CAyxNf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B
;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBx9cG,MAAO,KlBw9cO,QkBx9cP,El
Bw9ciB,CkBx9cjB,C;;QlB09cd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBr+cA,iB;MlBq+cA,sC;QAaiB,
QAaiB,Q;QAFb,IAryNO,qBAAQ,CAqyNf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF
,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB5+cG,MAAO,KlB4+cO,Qk
B5+cP,ElB4+ciB,CkB5+cjB,C;;QlB8+cd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBz/cA,iB;MlBy/cA,s
C;QAaiB,Q;QAFb,IAjzNO,qBAAQ,CAizNf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QAC
F,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBhgdG,MAAO,KlBggdO,
QkBhgdP,ElBggdiB,CkBhgdjB,C;;QlBkgdd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB7gdA,iB;MlB6gdA,sC;QAaiB,
QAaiB,Q;QAFb,IA7zNO,qBAAQ,CA6zNf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QAC
;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBphdG,MAAO,KlBo
hdO,QkBphdP,ElBohdiB,CkBphdjB,C;;QlBshdd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,oC;MAAA,8D;MkBj
idA,iB;MlBiidA,sC;QAaiB,Q;QAFb,IAz0NO,qBAAQ,CAy0Nf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,C
AAL,EAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,EAAT,C;UACR,WkBxidG
,MAAO,KlBwidO,QkBxidP,ElBwidiB,CkBxidjB,C;;QlB0idd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MA
AA,sC;QAWiB,Q;QAFb,IA35NO,qBAAQ,CA25Nf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,
C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,
KAAJ,C;YACI,WAAW,C;;;QAGnB,oB,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IAv6
NO,qBAAQ,CAu6Nf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAA
U,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,
C;;;QAGnB,oB,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IAn7NO,qBAAQ,CAm7
Nf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QA
AQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,oB,OAAO,Q
;KAjBX,C;+FAoBA,yB;MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IA/7NO,qBAAQ,CA+7N+Nf,C;UAAe,OAAO,I;QA
CtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,
CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,oB,OAAO,Q;O;KAjBX,C;+FAoBA,yB;
MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IA38NO,qBAAQ,CA28Nf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAA
K,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,
2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,oB,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MAAA,sC;
QAWiB,Q;QAFb,IAv9NO,qBAAQ,CAu9Nf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QAC
F,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,
C;YACI,WAAW,C;;;QAGnB,oB,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IAn+N
O,qBAAQ,CAm+Nf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CA
AV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;Q
AGnB,oB,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IA/+NO,qBAAQ,CA++N++Nf,C;
UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS
,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,oB,OAAO,Q;O;KAj
BX,C;+FAoBA,yB;MAAA,oC;MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IA3/NO,qBAAQ,CA2/Nf,C;UAAe,OAAO,I;Q
ACtB,eAAe,SAAS,sBAAK,CAAL,EAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,
EAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,oB,OAAO,Q;O;KAjBX,C;wFA

oBA,yB;MAAA,sE;MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IAjlOO,qBAAQ,CAilOf,C;UAAe,MAAM,6B;QACrB
,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CA
AT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,
OAAO,Q;O;KAnBX,C;0FAsBA,yB;MAAA,sE;MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IA/lOO,qBAAQ,CA+lOf,
C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QA
AQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C
;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;0FAsBA,yB;MAAA,sE;MAAA,8D;MAAA,kD;QAaiB,Q;QA
Fb,IA7mOO,qBAAQ,CA6mOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QA
Ab,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,C
AAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;0FAsBA,yB;MAAA,sE;MAA
A,8D;MAAA,kD;QAaiB,Q;QAFb,IA3nOO,qBAAQ,CA2nOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,C
AAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UA
AW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;
0FAsBA,yB;MAAA,sE;MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IAzoOO,qBAAQ,CAyoOf,C;UAAe,MAAM,6B;Q
ACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAA
L,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QA
GnB,OAAO,Q;O;KAnBX,C;0FAsBA,yB;MAAA,sE;MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IAvpOO,qBAAQ,C
AupOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UA
CI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,C
AAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;0FAsBA,yB;MAAA,sE;MAAA,8D;MAAA,kD;QAai
B,Q;QAFb,IArqOO,qBAAQ,CAqqOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+
B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAA
kB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;0FAsBA,yB;MAAA,sE;
MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IAnrOO,qBAAQ,CAmrOf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UA
AK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI
,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAn
BX,C;0FAsBA,yB;MAAA,sE;MAAA,oC;MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IAjsOO,qBAAQ,CAisOf,C;UA
Ae,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,EAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,S
AAS,sBAAK,CAAL,CAAT,EAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YA
CI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;oGAsBA,yB;MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IArxOO,qBA
AQ,CAqxOf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;
UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAA
tC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IAjy
OO,qBAAQ,CAiyOf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;
UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAA
tC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IAzzO
OO,qBAAQ,CAyzOf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;
UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAA
tC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IAr0O
OO,qBAAQ,CAq0Of,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;
UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAA
tC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAaiB,Q;QAFb,IA11O
OO,qBAAQ,CA01Of,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;
UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAA
tC,C;YACI,WAAW,C;;;Q

AGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IA71OO,qBAAQ,CA61Of,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAz2OO,qBAAQ,CAy2Of,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAGX,C;sGAoBA,yB;MAAA,oC;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAr3OO,qBAAQ,CAq3Of,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAGX,C;sGAoBA,yB;MAAA,oC;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAv8OO,qBAAQ,CAu8Of,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAGX,C;sGAoBA,yB;MAAA,oC;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAv9OO,qBAAQ,CAu9Of,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAGX,C;sGAoBA,yB;MAAA,oC;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAr+OO,qBAAQ,CAq+Of,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAGX,C;sGAoBA,yB;MAAA,oC;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAn/OO,qBAAQ,CAm/Of,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAGX,C;sGAoBA,yB;MAAA,oC;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAj/OO,qBAAQ,CAi/Of,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAGX,C;sGAoBA,yB;MAAA,oC;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAv/OO,qBAAQ,CAu/

CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA7qPO,qBAAQ,CA6qPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAnrPO,qBAAQ,CAmrPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAzrPO,qBAAQ,CAyrPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA/rPO,qBAAQ,CA+rPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IArsPO,qBAAQ,CAqsPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA3sPO,qBAAQ,CA2sPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAjtPO,qBAAQ,CAitPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAvtPO,qBAAQ,CAutPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA7tPO,qBAAQ,CA6tPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAnuPO,qBAAQ,CAmuPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAzuPO,qBAAQ,CAyuPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA/uPO,qBAAQ,CA+uPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IArvPO,qBAAQ,CAqvPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA3vPO,qBAAQ,CA2vPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAjwPO,qBAAQ,CAiwPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAvwPO,qBAAQ,CAuwPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA7wPO,qBAAQ,CA6wPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAnxPO,qBAAQ,CAmxPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAzxPO,qBAAQ,CAyxPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA/xPO,qBAAQ,CA+xPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IArzPO,qBAAQ,CAqzPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA3zPO,qBAAQ,CA2zPf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAj0PO,qBAAQ,CAi0Pf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAv0PO,qBAAQ,CAu0Pf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA70PO,qBAAQ,CA60Pf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAn1PO,qBAAQ,CAm1PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAz1PO,qBAAQ,CAy1PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA/1PO,qBAAQ,CA+1PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAr2PO,qBAAQ,CAq2PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA32PO,qBAAQ,CA22PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAj3PO,qBAAQ,CAi3PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAv3PO,qBAAQ,CAu3PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA73PO,qBAAQ,CA63PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAn4PO,qBAAQ,CAm4PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAz4PO,qBAAQ,CAy4PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA/4PO,qBAAQ,CA+4PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAr5PO,qBAAQ,CAq5PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA35PO,qBAAQ,CA25PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAj6PO,qBAAQ,CAi6PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAv6PO,qBAAQ,CAu6PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA76PO,qBAAQ,CA66PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAn7PO,qBAAQ,CAm7PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAz7PO,qBAAQ,CAy7PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA/7PO,qBAAQ,CA+7PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAr8PO,qBAAQ,CAq8PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA38PO,qBAAQ,CA28PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAj9PO,qBAAQ,CAi9PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAv9PO,qBAAQ,CAu9PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA79PO,qBAAQ,CA69PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAn+PO,qBAAQ,CAm+PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAz+PO,qBAAQ,CAy+PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA/+PO,qBAAQ,CA++PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAr/PO,qBAAQ,CAq/PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA3/PO,qBAAQ,CA2/PF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAjgQO,qBAAQ,CAigQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAvgQO,qBAAQ,CAugQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA7gQO,qBAAQ,CA6gQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAnhQO,qBAAQ,CAmhQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAzhQO,qBAAQ,CAyhQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA/hQO,qBAAQ,CA+hQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAriQO,qBAAQ,CAqiQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA3iQO,qBAAQ,CA2iQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAjjQO,qBAAQ,CAijQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAvjQO,qBAAQ,CAujQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA7jQO,qBAAQ,CA6jQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAnkQO,qBAAQ,CAmkQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAzkQO,qBAAQ,CAykQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA/kQO,qBAAQ,CA+kQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IArlQO,qBAAQ,CAqlQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA3lQO,qBAAQ,CA2lQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAjmQO,qBAAQ,CAimQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAvmQO,qBAAQ,CAumQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA7mQO,qBAAQ,CA6mQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAnnQO,qBAAQ,CAmnQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAznQO,qBAAQ,CAynQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA/nQO,qBAAQ,CA+nQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAroQO,qBAAQ,CAqoQF,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAA

I,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QA9LP,yB;O;KAHJ,C;kFAMA,yB;MA8LA,8D;MAAA,oC;MA9LA,sC;QAGW,sB;;UAkMP,IA3/PO,qBAAQ,CA2/Pf,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,UAAK,CAAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAtMmB,QAAsMJ,CAAAS,oBAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,UAAK,CAAAL,C;YACR,QAAzMe,QAAyMP,CAAAS,cAAT,C;YAAR,IAAI,2BAAW,CAAX,KAAY,KAAJ,C;cAAI,QAAJ,C;cAAI,QAAJ,C;;;UAGnB,qBAAO,O;;;QA/MP,/MP,yB;O;KAHJ,C;4FAMA,yB;MAAA,8D;MAAA,sC;QAOI,IAn4PO,qBAAQ,CAm4Pf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAAL,C;UACd,gBAAqB,cAAL,SAAK,C;UACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;UAAR,IAAI,2BAAW,CAAX,KAAY,KAAJ,C;YAAI,QAAJ,C;YAAI,QAAJ,C;;;QAGnB,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAAL,C;UACR,QAAzMe,QAAyMP,CAAAS,c

CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBpkfG,MA
AO,KlBokfO,QkBpkfP,ElBokfiB,CkBpkfjB,C;;QlBskfd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8
D;MkBjlfA,iB;MlBilfA,sC;QAeiB,Q;QAFb,IAloQO,qBAAQ,CAkoQf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,
UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,
WkB1lfG,MAAO,KlB0lfO,QkB1lfP,ElB0lfiB,CkB1lfjB,C;;QlB4lfd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,s
E;MAAA,8D;MkBvmfA,iB;MlBumfA,sC;QAeiB,Q;QAFb,IAhpQO,qBAAQ,CAgpQf,C;UAAe,MAAM,6B;QACr
B,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,C
AAT,C;UACR,WkBhnfG,MAAO,KlBgnfO,QkBhnfP,ElBgnfiB,CkBhnfjB,C;;QlBknfd,OAAO,Q;O;KAnBX,C;kF
AsBA,yB;MAAA,sE;MAAA,8D;MkB7nfA,iB;MlB6nfA,sC;QAeiB,Q;QAFb,IA9pQO,qBAAQ,CA8pQf,C;UAAe,
MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS
,UAAK,CAAL,CAAT,C;UACR,WkBtofG,MAAO,KlBsofO,QkBtofP,ElBsofiB,CkBtofjB,C;;QlBwofd,OAAO,Q;O
;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBnpfA,iB;MlBmpfA,sC;QAeiB,Q;QAFb,IA5qQO,qBAAQ,CA
4qQf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB5pfG,MAAO,KlB4pfO,QkB5pfP,ElB4pfiB,CkB5pfjB,C;;QlB
8pfd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBzqfA,iB;MlByqfA,sC;QAeiB,Q;QAFb,IA1r
QO,qBAAQ,CA0rQf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBlrfG,MAAO,KlBkrfO,QkBlrfP,ElBkrfiB,CkB
lrfjB,C;;QlBorfd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,oC;MAAA,8D;MkB/rfA,iB;MlB+rfA,
sC;QAeiB,Q;QAFb,IAxsQO,qBAAQ,CAwsQf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,EAAT,C;Q
ACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,EAAT,C;UACR,WkBxsfG,MAAO,KlBwsf
O,QkBxsfP,ElBwsfiB,CkBxsfjB,C;;QlB0sfd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAAA,8D;MkBhufA
,iB;MlBgufA,sC;QAeiB,Q;QAFb,IA9xQO,qBAAQ,CA8xQf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,C
AAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBzufG,
MAAO,KlByufO,QkBzufP,ElByufiB,CkBzufjB,C;;QlB2ufd,OAAO,Q;O;KAnBX,C;kFAsBA,yB;MAAA,sE;MAA
A,8D;MkBtvfA,iB;MlBsvfA,sC;QAeiB,Q;QAFb,IA5yQO,qBAAQ,CA4yQf,C;UAAe,MAAM,6B;QACrB,eAAe,S
AAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;U
ACR,WkB/vfG,MAAO,KlB+vfO,QkB/vfP,ElB+vfiB,CkB/vfjB,C;;QlBiwfd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;
MAAA,sE;MAAA,8D;MkB5wfA,iB;MlB4wfA,sC;QAeiB,Q;QAFb,IA1zQO,qBAAQ,CA0zQf,C;UAAe,MAAM,6
B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,
CAAT,C;UACR,WkBrxfG,MAAO,KlBqxfO,QkBrxfP,ElBqxfiB,CkBrxfjB,C;;QlBuxfd,OAAO,Q;O;KAnBX,C;mF
AsBA,yB;MAAA,sE;MAAA,8D;MkBlyfA,iB;MlBkyfA,sC;QAeiB,Q;QAFb,IA10QO,qBAAQ,CA00QX,C;UAAe,
MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS
,UAAK,CAAL,CAAT,C;UACR,WkB3yfG,MAAO,KlB2yfO,QkB3yfP,ElB2yfiB,CkB3yfjB,C;;QlB6yfd,OAAO,Q;O
;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBxzfA,iB;MlBwzfA,sC;QAeiB,Q;QAFb,IA51QO,qBAAQ,CA41
QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBj0fG,MAAO,KlBi0fO,QkBj0fP,ElBi0fiB,CkBj0fjB,C;;QlBm0f
d,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkB91fA,iB;MlB81fA,sC;QAeiB,Q;QAFb,IA92QO,qB
AAQ,CA82QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;
UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBv1fG,MAAO,KlBu1fO,QkBv1fP,ElBu1fiB,CkBv1fjB,C;;Q
lBy1fd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBp2fA,iB;MlBo2fA,sC;QAeiB,Q;QAFb,IAh4
QO,qBAAQ,CAg4Qf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB72fG,MAAO,KlB62fO,QkB72fP,ElB62fiB,CkB
72fjB,C;;QlB+2fd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,oC;MAAA,8D;MkB14fA,iB;MlB04fA,
sC;QAeiB,Q;QAFb,IAl5QO,qBAAQ,CAk5QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;Q
ACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBn5fG,MAAO,KlBm5f
O,QkBn5fP,ElBm5fiB,CkBn5fjB,C;;QlBq5fd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBh6fA
,iB;MlBg6fA,sC;QAeiB,Q;QAFb,IAp6QO,qBAAQ,CAo6QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,C
AAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBz6fG,
MAAO,KlBy6fO,QkBz6fP,ElBy6fiB,CkBz6fjB,C;;QlB26fd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAA
A,8D;MkBt7fA,iB;MlBs7fA,sC;QAeiB,Q;QAFb,IAt7QO,qBAAQ,CAs7QX,C;UAAe,MAAM,6B;QACrB,eAAe,S
AAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;U
ACR,WkB/7fG,MAAO,KlB+7fO,QkB/7fP,ElB+7fiB,CkB/7fjB,C;;QlBi8fd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;
MAAA,sE;MAAA,8D;MkB59fA,iB;MlB49fA,sC;QAeiB,Q;QAFb,IAx+QO,qBAAQ,CAw+QX,C;UAAe,MAAM,6
B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,
CAAT,C;UACR,WkBn+fG,MAAO,KlBm+fO,QkBn+fP,ElBm+fiB,CkBn+fjB,C;;QlBq+fd,OAAO,Q;O;KAnBX,C;mF
AsBA,yB;MAAA,sE;MAAA,8D;MkBhggBA,iB;MlBggBA,sC;QAeiB,Q;QAFb,IA1/QO,qBAAQ,CA0/QX,C;UAAe,
MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS
,UAAK,CAAL,CAAT,C;UACR,WkBvggBG,MAAO,KlBuggBO,QkBvggBP,ElBuggBiB,CkBvggBjB,C;;QlByggBd,OAAO,Q;O
;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBphgBA,iB;MlBohgBA,sC;QAeiB,Q;QAFb,IAhhQO,qBAAQ,CAg
hQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3hgBG,MAAO,KlB2hgBO,QkB3hgBP,ElB2hgBiB,CkB3hgBjB,C;;Q
lB6hgBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBxigBA,iB;MlBwigBA,sC;QAeiB,Q;QAFb,IAp
jQO,qBAAQ,CAojQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/igBG,MAAO,KlB+igBO,QkB/igBP,ElB+igBiB,CkB
/igBjB,C;;QlBijgBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkB5jgBA,iB;MlB4jgBA,sC;Q
AeiB,Q;QAFb,IAxkQO,qBAAQ,CAwkQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+
B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnkgBG,MAAO,KlBmkgBO,QkBnk
gBP,ElBmkgBiB,CkBnkgBjB,C;;QlBqkgBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBhlgBA,iB;
MlBglgBA,sC;QAeiB,Q;QAFb,IA5lQO,qBAAQ,CA4lQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,
CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvlgBG,MAAO,
KlBulgBO,QkBvlgBP,ElBulgBiB,CkBvlgBjB,C;;QlBylgBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D
;MkBpmgBA,iB;MlBomgBA,sC;QAeiB,Q;QAFb,IAhnQO,qBAAQ,CAgnQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS
,UAAK,CAAL,CAAT,C;UACR,WkB3mgBG,MAAO,KlB2mgBO,QkB3mgBP,ElB2mgBiB,CkB3mgBjB,C;;QlB6mgBd,OAAO,Q;O
;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBxngBA,iB;MlBwngBA,sC;QAeiB,Q;QAFb,IApoQO,qBAAQ,CAo
oQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/ngBG,MAAO,KlB+ngBO,QkB/ngBP,ElB+ngBiB,CkB/ngBjB,C;;Q
lBiogBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkB5ogBA,iB;MlB4ogBA,sC;QAeiB,Q;QAFb,IAx
pQO,qBAAQ,CAwpQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnpgBG,MAAO,KlBmpgBO,QkBnpgBP,ElBmpgBiB,CkB
npgBjB,C;;QlBqpgBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBhqgBA,iB;MlBgqgBA,sC;Q
AeiB,Q;QAFb,IA5qQO,qBAAQ,CA4qQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+
B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvqgBG,MAAO,KlBuqgBO,QkBvq
gBP,ElBuqgBiB,CkBvqgBjB,C;;QlByqgBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBprgBA,iB;
MlBorgBA,sC;QAeiB,Q;QAFb,IAhsQO,qBAAQ,CAgsQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,
CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3rgBG,MAAO,
Kl2rgBO,QkB3rgBP,El2rgBiB,CkB3rgBjB,C;;Ql6rgBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D
;MkBxsgBA,iB;MlBwsgBA,sC;QAeiB,Q;QAFb,IApsQO,qBAAQ,CAosQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAS
,UAAK,CAAL,CAAT,C;UACR,WkB/sgBG,MAAO,KlB+sgBO,QkB/sgBP,ElB+sgBiB,CkB/sgBjB,C;;QlBitgBd,OAAO,Q;O
;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkB5tgBA,iB;MlB4tgBA,sC;QAeiB,Q;QAFb,IAxtQO,qBAAQ,CAwt
QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnugBG,MAAO,KlBmugBO,QkBnugBP,ElBmugBiB,CkBnugBjB,C;;Q
lBqugBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBhvgBA,iB;MlBgvgBA,sC;QAeiB,Q;QAFb,IAp
uQO,qBAAQ,CAouQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvvgBG,MAAO,KlBuvgBO,QkBvvgBP,ElBuvgBiB,CkB
vvgBjB,C;;QlBywgBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBpwgBA,iB;MlBowgBA,sC;Q
AeiB,Q;QAFb,IAxxQO,qBAAQ,CAwxQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+
B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3wgBG,MAAO,KlB2wgBO,QkB3w
gBP,ElB2wgBiB,CkB3wgBjB,C;;QlB6wgBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBxxgBA,iB;
MlBwxgBA,sC;QAeiB,Q;QAFb,IApyQO,qBAAQ,CAoyQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,
CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/xgBG,MAAO,
Kl+xgBO,QkB/xgBP,El+xgBiB,CkB/xgBjB,C;;QlBiygBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D
;MkB5ygBA,iB;MlB4ygBA,sC;QAeiB,Q;QAFb,IAxzQO,qBAAQ,CAwzQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAS
,UAAK,CAAL,CAAT,C;UACR,WkBnzgBG,MAAO,KlBmzgBO,QkBnzgBP,ElBmzgBiB,CkBnzgBjB,C;;QlBqzgBd,OAAO,Q;O
;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBh0gBA,iB;MlBg0gBA,sC;QAeiB,Q;QAFb,IA50QO,qBAAQ,CA40
QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBv0gBG,MAAO,KlBu0gBO,QkBv0gBP,ElBu0gBiB,CkBv0gBjB,C;;Q
lBy0gBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBp1gBA,iB;MlBo1gBA,sC;QAeiB,Q;QAFb,IAh3
QO,qBAAQ,CAg3QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB31gBG,MAAO,KlB21gBO,QkB31gBP,ElB21gBiB,CkB
31gBjB,C;;QlB61gBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBx2gBA,iB;MlBw2gBA,sC;Q
AeiB,Q;QAFb,IA53QO,qBAAQ,CA43QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+
B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB32gBG,MAAO,KlB22gBO,QkB32
gBP,ElB22gBiB,CkB32gBjB,C;;QlB62gBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBx3gBA,iB;
MlBw3gBA,sC;QAeiB,Q;QAFb,IAp4QO,qBAAQ,CAo4QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,
CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/3gBG,MAAO,
Kl+3gBO,QkB/3gBP,El+3gBiB,CkB/3gBjB,C;;QliAgBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D
;MkB54gBA,iB;MlB44gBA,sC;QAeiB,Q;QAFb,IAx5QO,qBAAQ,CAw5QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAS
,UAAK,CAAL,CAAT,C;UACR,WkBn5gBG,MAAO,KlBm5gBO,QkBn5gBP,ElBm5gBiB,CkBn5gBjB,C;;QlBq5gBd,OAAO,Q;O
;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBh6gBA,iB;MlBg6gBA,sC;QAeiB,Q;QAFb,IA56QO,qBAAQ,CA46
QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBv6gBG,MAAO,KlBu6gBO,QkBv6gBP,ElBu6gBiB,CkBv6gBjB,C;;Q
lBy6gBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBp7gBA,iB;MlBo7gBA,sC;QAeiB,Q;QAFb,IAh8
QO,qBAAQ,CAg8QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB37gBG,MAAO,KlB27gBO,QkB37gBP,ElB27gBiB,CkB
37gBjB,C;;QlB67gBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBx8gBA,iB;MlBw8gBA,sC;Q
AeiB,Q;QAFb,IA58QO,qBAAQ,CA48QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+
B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB38gBG,MAAO,KlB28gBO,QkB38
gBP,ElB28gBiB,CkB38gBjB,C;;QlB68gBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBx9gBA,iB;
MlBw9gBA,sC;QAeiB,Q;QAFb,IAp+QO,qBAAQ,CAo+QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,
CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/9gBG,MAAO,
Kl+9gBO,QkB/9gBP,El+9gBiB,CkB/9gBjB,C;;QliBgeAd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D
;MkB5+gBA,iB;MlB4+gBA,sC;QAeiB,Q;QAFb,IAx/QO,qBAAQ,CAw/QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAS
,UAAK,CAAL,CAAT,C;UACR,WkBn/gBG,MAAO,KlBm/gBO,QkBn/gBP,ElBm/gBiB,CkBn/gBjB,C;;QlBq/gBd,OAAO,Q;O
;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBhghBA,iB;MlBghBA,sC;QAeiB,Q;QAFb,IA5gQO,qBAAQ,CA4g
QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvghBG,MAAO,KlBughBO,QkBvghBP,ElBughBiB,CkBvghBjB,C;;Q
lByghBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBpihBA,iB;MlBoihBA,sC;QAeiB,Q;QAFb,IAhi
QO,qBAAQ,CAgiQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3ihBG,MAAO,KlB2ihBO,QkB3ihBP,ElB2ihBiB,CkB
3ihBjB,C;;QlB6ihBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBxjhBA,iB;MlBwjhBA,sC;Q
AeiB,Q;QAFb,IA5jQO,qBAAQ,CA4jQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+
B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3jhBG,MAAO,KlB2jhBO,QkB3j
hBP,ElB2jhBiB,CkB3jhBjB,C;;QlB6jhBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBxkhBA,iB;
MlBwkhBA,sC;QAeiB,Q;QAFb,IAplQO,qBAAQ,CAolQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,
CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/khBG,MAAO,
Kl+khBO,QkB/khBP,El+khBiB,CkB/khBjB,C;;QliBgmAd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D
;MkB5lhBA,iB;MlB4lhBA,sC;QAeiB,Q;QAFb,IAxmQO,qBAAQ,CAwmQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAS
,UAAK,CAAL,CAAT,C;UACR,WkBnmhBG,MAAO,KlBmmhBO,QkBnmhBP,ElBmmhBiB,CkBnmhBjB,C;;QlBqmhBd,OAAO,Q;O
;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBhnhBA,iB;MlBgnhBA,sC;QAeiB,Q;QAFb,IA5nQO,qBAAQ,CA4n
QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvnhBG,MAAO,KlBunhBO,QkBvnhBP,ElBunhBiB,CkBvnhBjB,C;;Q
lBynhBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBpohBA,iB;MlBoohBA,sC;QAeiB,Q;QAFb,IAhp
QO,qBAAQ,CAgpQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3ohBG,MAAO,KlB2ohBO,QkB3ohBP,ElB2ohBiB,CkB
3ohBjB,C;;QlB6ohBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBxphBA,iB;MlBwphBA,sC;Q
AeiB,Q;QAFb,IA5pQO,qBAAQ,CA4pQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+
B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3phBG,MAAO,KlB2phBO,QkB3p
hBP,ElB2phBiB,CkB3phBjB,C;;QlB6phBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBxqhBA,iB;
MlBwqhBA,sC;QAeiB,Q;QAFb,IAprQO,qBAAQ,CAorQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,
CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/qhBG,MAAO,
Kl+qhBO,QkB/qhBP,El+qhBiB,CkB/qhBjB,C;;QliBgtAd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D
;MkB5rhBA,iB;MlB4rhBA,sC;QAeiB,Q;QAFb,IAxsQO,qBAAQ,CAwsQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAS
,UAAK,CAAL,CAAT,C;UACR,WkBnshBG,MAAO,KlBmshBO,QkBnshBP,ElBmshBiB,CkBnshBjB,C;;QlBqshBd,OAAO,Q;O
;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBhthBA,iB;MlBgthBA,sC;QAeiB,Q;QAFb,IA5tQO,qBAAQ,CA4t
QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvthBG,MAAO,KlButhBO,QkBvthBP,ElButhBiB,CkBvthBjB,C;;Q
lBythBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBpuhBA,iB;MlBouhBA,sC;QAeiB,Q;QAFb,IAhv
QO,qBAAQ,CAgvQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3uhBG,MAAO,KlB2uhBO,QkB3uhBP,ElB2uhBiB,CkB
3uhBjB,C;;QlB6uhBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBxvhBA,iB;MlBwvhBA,sC;Q
AeiB,Q;QAFb,IA5vQO,qBAAQ,CA4vQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+
B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3vhBG,MAAO,KlB2vhBO,QkB3v
hBP,ElB2vhBiB,CkB3vhBjB,C;;QlB6vhBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBxwhBA,iB;
MlBwwhBA,sC;QAeiB,Q;QAFb,IApxQO,qBAAQ,CAoxQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,
CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/whBG,MAAO,
Kl+whBO,QkB/whBP,El+whBiB,CkB/whBjB,C;;QliBgyAd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D
;MkB5xhBA,iB;MlB4xhBA,sC;QAeiB,Q;QAFb,IAxyQO,qBAAQ,CAwyQX,C;UAAe,MAAM,6B;QACrB,eAAe,SAS
,UAAK,CAAL,CAAT,C;UACR,WkBnyhBG,MAAO,KlBmyhBO,QkBnyhBP,ElBmyhBiB,CkBnyhBjB,C;;QlBqyhBd,OAAO,Q;O
;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBhzhBA,iB;MlBgzhBA,sC;QAeiB,Q;QAFb,IA5zQO,qBAAQ,CA4z
QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI
,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvzhBG,MAAO,KlBuzhBO,QkBvzhBP,ElBuzhBiB,CkBvzhBjB,C;;Q
lByzhBd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MkBp0hBA,iB;MlBo0hBA,sC;QAeiB,Q;QAFb,IAh1
QO,qBAAQ,CAg1QX,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,
CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UAC

L,EAAT,C;UACR,WkBz5fG,MAAO,KlBy5fO,QkBz5fP,ElBy5fiB,CkBz5fjB,C;;QlB25fd,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAl+QO,qBAAQ,CAk+Qf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAh/QO,qBAAQ,CAg/Qf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA9/QO,qBAAQ,CA8/Qf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA5gRO,qBAAQ,CA4gRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IA1hRO,qBAAQ,CA0hRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAxiRO,qBAAQ,CAwiRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAtjRO,qBAAQ,CAsjRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IApkRO,qBAAQ,CAokRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAllRO,qBAAQ,CAklRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnmRO,qBAAQ,CAmmRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnnRO,qBAAQ,CAmnRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAtoRO,qBAAQ,CAsoRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IApqRO,qBAAQ,CAoqRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAlrRO,qBAAQ,CAkrRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnsRO,qBAAQ,CAmsRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAntRO,qBAAQ,CAmtRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAvuRO,qBAAQ,CAuuRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnvRO,qBAAQ,CAmvRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnwRO,qBAAQ,CAmwRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnxRO,qBAAQ,CAmxRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnyRO,qBAAQ,CAmyRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAnzRO,qBAAQ,CAmzRf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn0RO,qBAAQ,CAm0Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn1RO,qBAAQ,CAm1Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn2RO,qBAAQ,CAm2Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn3RO,qBAAQ,CAm3Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn4RO,qBAAQ,CAm4Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn5RO,qBAAQ,CAm5Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn6RO,qBAAQ,CAm6Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn7RO,qBAAQ,CAm7Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn8RO,qBAAQ,CAm8Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn9RO,qBAAQ,CAm9Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn+RO,qBAAQ,CAm+Rf,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAnBX,C;mFAsBA,yB;MAAA,sE;MAAA,8D;MAAA,sC;QAaiB,Q;QAFb,IAn/RO,qBAAQ,CAm/

SAAS,UAAK,CAAL,CAAT,C;UACR,WkBhugBG,MAAO,KlBgugBO,QkBhugBP,ElBgugBiB,CkBhugBjB,C;;QlBkugBd,OAAO,Q;O;KAjBX,C;8FAoBA,yB;MAAA,8D;MkB7ugBA,iB;MlB6ugBA,sC;QAaiB,Q;QAFb,IA5vRO,qBAAQ,CA4vRf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBpvgBG,MAAO,KlBovgBO,QkBpvgBP,ElBovgBiB,CkBpvgBjB,C;;QlBsvgBd,OAAO,Q;O;KAjBX,C;8FAoBA,yB;MAAA,oC;MAAA,8D;MkBjwgBA,iB;MlBiwgBA,sC;QAaiB,Q;QAFb,IAxwRO,qBAAQ,CAAwwRf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,EAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,EAAT,C;UACR,WkBxwgBG,MAAO,KlBwwgBO,QkBxwgBP,ElBwwgBiB,CkBxwgBjB,C;;QlB0wgBd,OAAO,Q;O;KAjBX,C;8FAoBA,yB;MAAA,8D;MkBhygBA,iB;MlBgygBA,sC;QAaiB,Q;QAFb,IA51RO,qBAAQ,CA41Rf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvygBG,MAAO,KlBuygBO,QkBvygBP,ElBuygBiB,CkBvygBjB,C;;QlBuygBd,OAAO,Q;O;KAjBX,C;8FAoBA,yB;MAAA,8D;MkBp0gBA,iB;MlBo0gBA,sC;QAaiB,Q;QAFb,IAx2RO,qBAAQ,CAAw2Rf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3zgBG,MAAO,KlB2zgBO,QkB3zgBP,ElB2zgBiB,CkB3zgBjB,C;;QlB6zgBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBx1gBA,iB;MlBw1gBA,sC;QAaiB,Q;QAFb,IAp3RO,qBAAQ,CAAo3Rf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/1gBG,MAAO,KlB+1gBO,QkB/1gBP,ElB+1gBiB,CkB/1gBjB,C;;QlBi2gBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB54gBA,iB;MlB44gBA,sC;QAaiB,Q;QAFb,IAh5RO,qBAAQ,CAAg5Rf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBn5gBG,MAAO,KlBm5gBO,QkBn5gBP,ElBm5gBiB,CkBn5gBjB,C;;QlBq5gBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBh7gBA,iB;MlBg7gBA,sC;QAaiB,Q;QAFb,IA56RO,qBAAQ,CAA46Rf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBv7gBG,MAAO,KlBu7gBO,QkBv7gBP,ElBu7gBiB,CkBv7gBjB,C;;QlBy7gBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBp9gBA,iB;MlBo9gBA,sC;QAaiB,Q;QAFb,IAx7RO,qBAAQ,CAAw7Rf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB39gBG,MAAO,KlB29gBO,QkB39gBP,ElB29gBiB,CkB39gBjB,C;;QlB69gBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBx/gBA,iB;MlBw/gBA,sC;QAaiB,Q;QAFb,IA48RO,qBAAQ,CA48Rf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB//gBG,MAAO,KlB+/gBO,QkB//gBP,ElB+/gBiB,CkB//gBjB,C;;QlBihBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB5hhBA,iB;MlB4hhBA,sC;QAaiB,Q;QAFb,IAh+RO,qBAAQ,CAAg+Rf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnhhBG,MAAO,KlBmhhBO,QkBnhhBP,ElBmhhBiB,CkBnhhBjB,C;;QlBqhhBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhjhBA,iB;MlBgjhBA,sC;QAaiB,Q;QAFb,IAo/RO,qBAAQ,CAAo/Rf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBvjhBG,MAAO,KlBujhBO,QkBvjhBP,ElBujhBiB,CkBvjhBjB,C;;QlBykhBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBpmhBA,iB;MlBomhBA,sC;QAaiB,Q;QAFb,IAwgSO,qBAAQ,CAwgSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3mhBG,MAAO,KlB2mhBO,QkB3mhBP,ElB2mhBiB,CkB3mhBjB,C;;QlB6mhBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxohBA,iB;MlBwohBA,sC;QAaiB,Q;QAFb,IA4hSO,qBAAQ,CA4hSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/ohBG,MAAO,KlB+ohBO,QkB/ohBP,ElB+ohBiB,CkB/ohBjB,C;;QlBiphBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB5qhBA,iB;MlB4qhBA,sC;QAaiB,Q;QAFb,IAojSO,qBAAQ,CAojSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnrhBG,MAAO,KlBmrhBO,QkBnrhBP,ElBmrhBiB,CkBnrhBjB,C;;QlBqrhBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhthBA,iB;MlBgthBA,sC;QAaiB,Q;QAFb,IAwkSO,qBAAQ,CAwkSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3thBG,MAAO,KlB2thBO,QkB3thBP,ElB2thBiB,CkB3thBjB,C;;QlB6thBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxvhBA,iB;MlBwvhBA,sC;QAaiB,Q;QAFb,IAolSO,qBAAQ,CAolSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/vhBG,MAAO,KlB+vhBO,QkB/vhBP,ElB+vhBiB,CkB/vhBjB,C;;QlBiwhBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB5xhBA,iB;MlB4xhBA,sC;QAaiB,Q;QAFb,IA4mSO,qBAAQ,CA4mSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnyhBG,MAAO,KlBmyhBO,QkBnyhBP,ElBmyhBiB,CkBnyhBjB,C;;QlBqyhBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBh0hBA,iB;MlBg0hBA,sC;QAaiB,Q;QAFb,IAwpSO,qBAAQ,CAwpSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB30hBG,MAAO,KlB20hBO,QkB30hBP,ElB20hBiB,CkB30hBjB,C;;QlB60hBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBx2hBA,iB;MlBw2hBA,sC;QAaiB,Q;QAFb,IAorSO,qBAAQ,CAorSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/2hBG,MAAO,KlB+2hBO,QkB/2hBP,ElB+2hBiB,CkB/2hBjB,C;;QlBi3hBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB54hBA,iB;MlB44hBA,sC;QAaiB,Q;QAFb,IA4tSO,qBAAQ,CA4tSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBn5hBG,MAAO,KlBm5hBO,QkBn5hBP,ElBm5hBiB,CkBn5hBjB,C;;QlBq5hBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBh7hBA,iB;MlBg7hBA,sC;QAaiB,Q;QAFb,IAwyS,qBAAQ,CAwyS,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB37hBG,MAAO,KlB27hBO,QkB37hBP,ElB27hBiB,CkB37hBjB,C;;QlB67hBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBx9hBA,iB;MlBw9hBA,sC;QAaiB,Q;QAFb,IA4zS,qBAAQ,CA4zS,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/9hBG,MAAO,KlB+9hBO,QkB/9hBP,ElB+9hBiB,CkB/9hBjB,C;;QlBi+hBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB5/hBA,iB;MlB4/hBA,sC;QAaiB,Q;QAFb,IA42S,qBAAQ,CA42S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBngiBG,MAAO,KlBmgiBO,QkBngiBP,ElBmgiBiB,CkBngiBjB,C;;QlBqgiBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhiiBA,iB;MlBgiiBA,sC;QAaiB,Q;QAFb,IAw3S,qBAAQ,CAw3S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3iiBG,MAAO,KlB2iiBO,QkB3iiBP,ElB2iiBiB,CkB3iiBjB,C;;QlB6iiBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxkiBA,iB;MlBwkiBA,sC;QAaiB,Q;QAFb,IA44S,qBAAQ,CA44S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/kiBG,MAAO,KlB+kiBO,QkB/kiBP,ElB+kiBiB,CkB/kiBjB,C;;QlBimiBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB5miBA,iB;MlB4miBA,sC;QAaiB,Q;QAFb,IA45S,qBAAQ,CA45S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnniBG,MAAO,KlBmniBO,QkBnniBP,ElBmniBiB,CkBnniBjB,C;;QlBqniBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhpiBA,iB;MlBgpiBA,sC;QAaiB,Q;QAFb,IAw5S,qBAAQ,CAw5S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3piBG,MAAO,KlB2piBO,QkB3piBP,ElB2piBiB,CkB3piBjB,C;;QlB6piBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxriBA,iB;MlBwriBA,sC;QAaiB,Q;QAFb,IA46S,qBAAQ,CA46S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/riBG,MAAO,KlB+riBO,QkB/riBP,ElB+riBiB,CkB/riBjB,C;;QlBismBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB5smBA,iB;MlB4smBA,sC;QAaiB,Q;QAFb,IA47S,qBAAQ,CA47S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBntmBG,MAAO,KlBmtmBO,QkBntmBP,ElBmtmBiB,CkBntmBjB,C;;QlBqtmBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhvmBA,iB;MlBgvmBA,sC;QAaiB,Q;QAFb,IAw8S,qBAAQ,CAw8S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3vmBG,MAAO,KlB2vmBO,QkB3vmBP,ElB2vmBiB,CkB3vmBjB,C;;QlB6vmBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxxmBA,iB;MlBwxmBA,sC;QAaiB,Q;QAFb,IA49S,qBAAQ,CA49S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/xmBG,MAAO,KlB+xmBO,QkB/xmBP,ElB+xmBiB,CkB/xmBjB,C;;QlBizmBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB50mBA,iB;MlB40mBA,sC;QAaiB,Q;QAFb,IA4+S,qBAAQ,CA4+S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBn1mBG,MAAO,KlBm1mBO,QkBn1mBP,ElBm1mBiB,CkBn1mBjB,C;;QlBq1mBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBh3mBA,iB;MlBg3mBA,sC;QAaiB,Q;QAFb,IAw/S,qBAAQ,CAw/S,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB33mBG,MAAO,KlB23mBO,QkB33mBP,ElB23mBiB,CkB33mBjB,C;;QlB63mBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBx5mBA,iB;MlBw5mBA,sC;QAaiB,Q;QAFb,IA4gT,qBAAQ,CA4gT,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/5mBG,MAAO,KlB+5mBO,QkB/5mBP,ElB+5mBiB,CkB/5mBjB,C;;QlBi6mBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB57mBA,iB;MlB47mBA,sC;QAaiB,Q;QAFb,IAohT,qBAAQ,CAohT,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBn8mBG,MAAO,KlBm8mBO,QkBn8mBP,ElBm8mBiB,CkBn8mBjB,C;;QlBq8mBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBh+mBA,iB;MlBg+mBA,sC;QAaiB,Q;QAFb,IAwiT,qBAAQ,CAwiT,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3+mBG,MAAO,KlB2+mBO,QkB3+mBP,ElB2+mBiB,CkB3+mBjB,C;;QlB6+mBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxgnBA,iB;MlBwgnBA,sC;QAaiB,Q;QAFb,IA4jT,qBAAQ,CA4jT,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB/gnBG,MAAO,KlB+gnBO,QkB/gnBP,ElB+gnBiB,CkB/gnBjB,C;;QlBiinBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB5inBA,iB;MlB4inBA,sC;QAaiB,Q;QAFb,IAokT,qBAAQ,CApkTf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnjnBG,MAAO,KlBmjnBO,QkBnjnBP,ElBmjnBiB,CkBnjnBjB,C;;QlBqjnBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhlnBA,iB;MlBglnBA,sC;QAaiB,Q;QAFb,IAwlkB,qBAAQ,CAxlkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3lnBG,MAAO,KlB2lnBO,QkB3lnBP,ElB2lnBiB,CkB3lnBjB,C;;QlB6lnBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxnnBA,iB;MlBwnnBA,sC;QAaiB,Q;QAFb,IA4mkB,qBAAQ,CA5mkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnnnBG,MAAO,KlBmnnBO,QkBnnnBP,ElBmnnBiB,CkBnnnBjB,C;;QlBqnnBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhpnBA,iB;MlBgpnBA,sC;QAaiB,Q;QAFb,IAwnkB,qBAAQ,CAxnkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3pnBG,MAAO,KlB2pnBO,QkB3pnBP,ElB2pnBiB,CkB3pnBjB,C;;QlB6pnBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxrnBA,iB;MlBwrnBA,sC;QAaiB,Q;QAFb,IA4okB,qBAAQ,CA5okBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnrnBG,MAAO,KlBmrnBO,QkBnrnBP,ElBmrnBiB,CkBnrnBjB,C;;QlBqrnBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhtnBA,iB;MlBgtnBA,sC;QAaiB,Q;QAFb,IAwpkB,qBAAQ,CAxpkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3tnBG,MAAO,KlB2tnBO,QkB3tnBP,ElB2tnBiB,CkB3tnBjB,C;;QlB6tnBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxvnBA,iB;MlBwvnBA,sC;QAaiB,Q;QAFb,IA4qkB,qBAAQ,CA5qkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnvnBG,MAAO,KlBmvnBO,QkBnvnBP,ElBmvnBiB,CkBnvnBjB,C;;QlBqvnBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhxnBA,iB;MlBgxnBA,sC;QAaiB,Q;QAFb,IAwrkB,qBAAQ,CAxrkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3xnBG,MAAO,KlB2xnBO,QkB3xnBP,ElB2xnBiB,CkB3xnBjB,C;;QlB6xnBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxznBA,iB;MlBwznBA,sC;QAaiB,Q;QAFb,IA4skB,qBAAQ,CA5skBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnznBG,MAAO,KlBmznBO,QkBnznBP,ElBmznBiB,CkBnznBjB,C;;QlBqznBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBh1nBA,iB;MlBg1nBA,sC;QAaiB,Q;QAFb,IAwtkB,qBAAQ,CAxtkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB31nBG,MAAO,KlB21nBO,QkB31nBP,ElB21nBiB,CkB31nBjB,C;;QlB61nBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBx3nBA,iB;MlBw3nBA,sC;QAaiB,Q;QAFb,IA4ukB,qBAAQ,CA5ukBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBn3nBG,MAAO,KlBm3nBO,QkBn3nBP,ElBm3nBiB,CkBn3nBjB,C;;QlBq3nBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBh5nBA,iB;MlBg5nBA,sC;QAaiB,Q;QAFb,IAwvkB,qBAAQ,CAxvkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB35nBG,MAAO,KlB25nBO,QkB35nBP,ElB25nBiB,CkB35nBjB,C;;QlB65nBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBx7nBA,iB;MlBw7nBA,sC;QAaiB,Q;QAFb,IA4wkB,qBAAQ,CA5wkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBn7nBG,MAAO,KlBm7nBO,QkBn7nBP,ElBm7nBiB,CkBn7nBjB,C;;QlBq7nBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBh9nBA,iB;MlBg9nBA,sC;QAaiB,Q;QAFb,IAwxkB,qBAAQ,CAxxkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB39nBG,MAAO,KlB29nBO,QkB39nBP,ElB29nBiB,CkB39nBjB,C;;QlB69nBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBx/nBA,iB;MlBw/nBA,sC;QAaiB,Q;QAFb,IA4ykB,qBAAQ,CA5ykBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBn/nBG,MAAO,KlBm/nBO,QkBn/nBP,ElBm/nBiB,CkBn/nBjB,C;;QlBq/nBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhhoBA,iB;MlBghoBA,sC;QAaiB,Q;QAFb,IAwzkB,qBAAQ,CAxzkBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3hoBG,MAAO,KlB2hoBO,QkB3hoBP,ElB2hoBiB,CkB3hoBjB,C;;QlB6hoBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxjoBA,iB;MlBwjoBA,sC;QAaiB,Q;QAFb,IA40kB,qBAAQ,CA50kBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBnjoBG,MAAO,KlBmjoBO,QkBnjoBP,ElBmjoBiB,CkBnjoBjB,C;;QlBqjoBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBhloBA,iB;MlBgloBA,sC;QAaiB,Q;QAFb,IAw1kB,qBAAQ,CAx1kBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB3loBG,MAAO,KlB2loBO,QkB3loBP,ElB2loBiB,CkB3loBjB,C;;QlB6loBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBxnoBA,iB;MlBwnoBA,sC;QAaiB,Q;QAFb,IA42kB,qBAAQ,CA52kBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WknoBG,MAAO,KlBknoBO,QkBlnoBP,ElBknoBiB,CkBlnoBjB,C;;QlBonoBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB/poBA,iB;MlB+poBA,sC;QAaiB,Q;QAFb,IAw3kB,qBAAQ,CAx3kBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB1poBG,MAAO,KlB0poBO,QkB1poBP,ElB0poBiB,CkB1poBjB,C;;QlB4poBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBvroBA,iB;MlBuroBA,sC;QAaiB,Q;QAFb,IA44kB,qBAAQ,CA54kBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBlroBG,MAAO,KlBkroBO,QkBlroBP,ElBkroBiB,CkBlroBjB,C;;QlBoroBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB/soBA,iB;MlB+soBA,sC;QAaiB,Q;QAFb,IAw5kB,qBAAQ,CAx5kBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkB1soBG,MAAO,KlB0soBO,QkB1soBP,ElB0soBiB,CkB1soBjB,C;;QlB4soBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkBvuoBA,iB;MlBuuoBA,sC;QAaiB,Q;QAFb,IA46kB,qBAAQ,CA56kBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,WkBluoBG,MAAO,KlBkuoBO,QkBluoBP,ElBkuoBiB,CkBluoBjB,C;;QlBouoBd,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MkB/voBA,iB;MlB+voBA,sC;QAaiB,Q;QAFb,IAw7kB,qBAAQ,CAx7kBf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,

CA8jSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IA1kSO,qBAAQ,CA0kSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IAtlSO,qBAAQ,CAslSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IAlmSO,qBAAQ,CAkmSf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;+FAoBA,yB;MAAA,8D;MAAA,sC;QAWiB,Q;QAFb,IA9mSO,qBAAQ,CA8m

UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAh6SO,qBAAQ,CAg6Sf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IA56SO,qBAAQ,CA46Sf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAx7SO,qBAAQ,CAw7Sf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IAp8SO,qBAAQ,CAo8Sf,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,UAAK,CAAL,CAAT,C;QACF,+B;QAAb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,UAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAjBX,C;sGAoBA,yB;MAAA,8D;MAAA,kD;QAWiB,Q;QAFb,IA59SO,qBAAQ,CA49

OAAO,2BAAc,UAAd,C;K;IAGX,0C;MAGI,OAAO,2BAAc,UAAd,C;K;IAGX,0C;MAGI,OAAO,2BAAc,UAAd,C;K;IAGX,0C;MAGI,OAAO,2BAAc,UAAd,C;K;IAGX,8C;MAOiB,Q;MAFb,IAlwTO,qBAAQ,CAkwTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,+B;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAxwTO,qBAAQ,CAwwTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA9wTO,qBAAQ,CA8wTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IApxTO,qBAAQ,CAoxTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA1xTO,qBAAQ,CA0xTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAhyTO,qBAAQ,CAgyTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAtyTO,qBAAQ,CAsyTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA5yTO,qBAAQ,CA4yTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAlzTO,qBAAQ,CAkzTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAxzTO,qBAAQ,CAwzTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA1zTO,qBAAQ,CA0zTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA5zTO,qBAAQ,CA4zTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA9zTO,qBAAQ,CA8zTf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh0TO,qBAAQ,CAg0Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt0TO,qBAAQ,CAs0Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA10TO,qBAAQ,CA00Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA50TO,qBAAQ,CA40Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA90TO,qBAAQ,CA80Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh1TO,qBAAQ,CAg1Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt1TO,qBAAQ,CAs1Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA11TO,qBAAQ,CA01Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA51TO,qBAAQ,CA41Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA91TO,qBAAQ,CA81Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh2TO,qBAAQ,CAg2Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt2TO,qBAAQ,CAs2Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA12TO,qBAAQ,CA02Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA52TO,qBAAQ,CA42Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA92TO,qBAAQ,CA82Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh3TO,qBAAQ,CAg3Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt3TO,qBAAQ,CAs3Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA13TO,qBAAQ,CA03Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA53TO,qBAAQ,CA43Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA93TO,qBAAQ,CA83Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh4TO,qBAAQ,CAg4Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt4TO,qBAAQ,CAs4Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA14TO,qBAAQ,CA04Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA54TO,qBAAQ,CA44Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA94TO,qBAAQ,CA84Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh5TO,qBAAQ,CAg5Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt5TO,qBAAQ,CAs5Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA15TO,qBAAQ,CA05Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA55TO,qBAAQ,CA45Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA95TO,qBAAQ,CA85Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh6TO,qBAAQ,CAg6Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt6TO,qBAAQ,CAs6Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA16TO,qBAAQ,CA06Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA56TO,qBAAQ,CA46Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA96TO,qBAAQ,CA86Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh7TO,qBAAQ,CAg7Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt7TO,qBAAQ,CAs7Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA17TO,qBAAQ,CA07Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA57TO,qBAAQ,CA47Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA97TO,qBAAQ,CA87Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh8TO,qBAAQ,CAg8Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt8TO,qBAAQ,CAs8Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA18TO,qBAAQ,CA08Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA58TO,qBAAQ,CA48Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA98TO,qBAAQ,CA88Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh9TO,qBAAQ,CAg9Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt9TO,qBAAQ,CAs9Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA19TO,qBAAQ,CA09Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA59TO,qBAAQ,CA49Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA99TO,qBAAQ,CA89Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh+TO,qBAAQ,CAg+Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt+TO,qBAAQ,CAs+Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA1+TO,qBAAQ,CA0+Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA5+TO,qBAAQ,CA4+Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA9+TO,qBAAQ,CA8+Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAh/TO,qBAAQ,CAg/Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAt/TO,qBAAQ,CAs/Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA1/TO,qBAAQ,CA0/Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA5/TO,qBAAQ,CA4/Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA9/TO,qBAAQ,CA8/Tf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAhhUO,qBAAQ,CAghUf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IAthUO,qBAAQ,CAshUf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,gD;MAOiB,Q;MAFb,IA1hUO,qBAAQ,CA0hUf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9

AP,C;;MAArC,gB;K;oFAGJ,yB;MAAA,oC;MAAA,gC;MAAA,oC;QAMmC,Q;QAAhB,iD;UAAgB,cAAhB,0B;UAAsB,OAAO,oBAAP,C;;QAArC,gB;O;KANJ,C;gGASA,6B;MArjJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA8jJnB,gB;K;kGAGJ,6B;MAvjJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgkJnB,gB;K;kGAGJ,6B;MAzjJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkkJnB,gB;K;kGAGJ,6B;MA3jJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAokJnB,gB;K;kGAGJ,6B;MA7jJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAskJnB,gB;K;kGAGJ,6B;MA/jJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwkJnB,gB;K;kGAGJ,6B;MAjkJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0kJnB,gB;K;kGAGJ,6B;MAnkJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4kJnB,gB;K;kGAGJ,6B;MArkJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAglJnB,gB;K;kGAGJ,6B;MAzkJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAklJnB,gB;K;kGAGJ,6B;MA3kJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAolJnB,gB;K;kGAGJ,6B;MA7kJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAslJnB,gB;K;kGAGJ,6B;MA/kJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwlJnB,gB;K;kGAGJ,6B;MAjlJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0lJnB,gB;K;kGAGJ,6B;MAnlJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4lJnB,gB;K;kGAGJ,6B;MArlJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgmJnB,gB;K;kGAGJ,6B;MAzlJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkmJnB,gB;K;kGAGJ,6B;MA3lJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAomJnB,gB;K;kGAGJ,6B;MA7lJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAsmJnB,gB;K;kGAGJ,6B;MA/lJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwmJnB,gB;K;kGAGJ,6B;MAjmJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0mJnB,gB;K;kGAGJ,6B;MAnmJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4mJnB,gB;K;kGAGJ,6B;MArmJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgnJnB,gB;K;kGAGJ,6B;MAzmJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAknJnB,gB;K;kGAGJ,6B;MA3mJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAonJnB,gB;K;kGAGJ,6B;MA7mJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAsnJnB,gB;K;kGAGJ,6B;MA/mJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwnJnB,gB;K;kGAGJ,6B;MAjnJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0nJnB,gB;K;kGAGJ,6B;MAnnJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4nJnB,gB;K;kGAGJ,6B;MArnJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgoJnB,gB;K;kGAGJ,6B;MAznJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkoJnB,gB;K;kGAGJ,6B;MA3nJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAooJnB,gB;K;kGAGJ,6B;MA7nJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAsoJnB,gB;K;kGAGJ,6B;MA/nJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwoJnB,gB;K;kGAGJ,6B;MAjoJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0oJnB,gB;K;kGAGJ,6B;MAnoJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4oJnB,gB;K;kGAGJ,6B;MAroJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgpJnB,gB;K;kGAGJ,6B;MAzoJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkpJnB,gB;K;kGAGJ,6B;MA3oJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAopJnB,gB;K;kGAGJ,6B;MA7oJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAspJnB,gB;K;kGAGJ,6B;MA/oJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwpJnB,gB;K;kGAGJ,6B;MAjpJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0pJnB,gB;K;kGAGJ,6B;MAnpJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4pJnB,gB;K;kGAGJ,6B;MArpJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgqJnB,gB;K;kGAGJ,6B;MAzpJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkqJnB,gB;K;kGAGJ,6B;MA3pJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAoqJnB,gB;K;kGAGJ,6B;MA7pJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAsqJnB,gB;K;kGAGJ,6B;MA/pJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwqJnB,gB;K;kGAGJ,6B;MAjqJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0qJnB,gB;K;kGAGJ,6B;MAnqJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4qJnB,gB;K;kGAGJ,6B;MArqJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgrJnB,gB;K;kGAGJ,6B;MAzqJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkrJnB,gB;K;kGAGJ,6B;MA3qJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAorJnB,gB;K;kGAGJ,6B;MA7qJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAsrJnB,gB;K;kGAGJ,6B;MA/qJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwrJnB,gB;K;kGAGJ,6B;MAjrJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0rJnB,gB;K;kGAGJ,6B;MAnrJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4rJnB,gB;K;kGAGJ,6B;MArrJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgsJnB,gB;K;kGAGJ,6B;MAzrJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAksJnB,gB;K;kGAGJ,6B;MA3rJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAosJnB,gB;K;kGAGJ,6B;MA7rJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAssJnB,gB;K;kGAGJ,6B;MA/rJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwsJnB,gB;K;kGAGJ,6B;MAjsJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0sJnB,gB;K;kGAGJ,6B;MAnsJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4sJnB,gB;K;kGAGJ,6B;MArsJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgtJnB,gB;K;kGAGJ,6B;MAztJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAktJnB,gB;K;kGAGJ,6B;MA3sJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAotJnB,gB;K;kGAGJ,6B;MA7sJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAstJnB,gB;K;kGAGJ,6B;MA/sJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwtJnB,gB;K;kGAGJ,6B;MAjtJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0tJnB,gB;K;kGAGJ,6B;MAntJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4tJnB,gB;K;kGAGJ,6B;MArtJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAguJnB,gB;K;kGAGJ,6B;MAztJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkuJnB,gB;K;kGAGJ,6B;MA3tJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAouJnB,gB;K;kGAGJ,6B;MA7tJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAsuJnB,gB;K;kGAGJ,6B;MA/tJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwuJnB,gB;K;kGAGJ,6B;MAjuJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0uJnB,gB;K;kGAGJ,6B;MAnuJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4uJnB,gB;K;kGAGJ,6B;MAruJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgvJnB,gB;K;kGAGJ,6B;MAzuJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkvJnB,gB;K;kGAGJ,6B;MA3uJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAovJnB,gB;K;kGAGJ,6B;MA7uJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAsvJnB,gB;K;kGAGJ,6B;MA/uJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwvJnB,gB;K;kGAGJ,6B;MAjvJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0vJnB,gB;K;kGAGJ,6B;MAnvJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4vJnB,gB;K;kGAGJ,6B;MArvJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgwJnB,gB;K;kGAGJ,6B;MAzvJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkwJnB,gB;K;kGAGJ,6B;MA3vJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAowJnB,gB;K;kGAGJ,6B;MA7vJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAswJnB,gB;K;kGAGJ,6B;MA/vJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwwJnB,gB;K;kGAGJ,6B;MAjwJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0wJnB,gB;K;kGAGJ,6B;MAnwJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4wJnB,gB;K;kGAGJ,6B;MArwJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgxJnB,gB;K;kGAGJ,6B;MAzwJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkxJnB,gB;K;kGAGJ,6B;MA3wJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAoxJnB,gB;K;kGAGJ,6B;MA7wJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAsxJnB,gB;K;kGAGJ,6B;MA/wJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwxJnB,gB;K;kGAGJ,6B;MAjxJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0xJnB,gB;K;kGAGJ,6B;MAnxJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4xJnB,gB;K;kGAGJ,6B;MArxJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgyJnB,gB;K;kGAGJ,6B;MAzxJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkyJnB,gB;K;kGAGJ,6B;MA3xJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAoyJnB,gB;K;kGAGJ,6B;MA7xJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAsyJnB,gB;K;kGAGJ,6B;MA/xJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwyJnB,gB;K;kGAGJ,6B;MAjyJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0yJnB,gB;K;kGAGJ,6B;MAnyJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4yJnB,gB;K;kGAGJ,6B;MAryJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAgzJnB,gB;K;kGAGJ,6B;MAzyJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAkzJnB,gB;K;kGAGJ,6B;MA3yJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAozJnB,gB;K;kGAGJ,6B;MA7yJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAszJnB,gB;K;kGAGJ,6B;MA/yJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAwzJnB,gB;K;kGAGJ,6B;MAjzJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA0zJnB,gB;K;kGAGJ,6B;MAnzJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA4zJnB,gB;K;kGAGJ,6B;MArzJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAg0JnB,gB;K;kGAGJ,6B;MAzzJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAk0JnB,gB;K;kGAGJ,6B;MA3zJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAo0JnB,gB;K;kGAGJ,6B;MA7zJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAs0JnB,gB;K;kGAGJ,6B;MA/zJiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAw0JnB,gB;K;kGAGJ,6B;MAj0JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA00JnB,gB;K;kGAGJ,6B;MAn0JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA40JnB,gB;K;kGAGJ,6B;MAr0JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAg1JnB,gB;K;kGAGJ,6B;MAz0JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAk1JnB,gB;K;kGAGJ,6B;MA30JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAo1JnB,gB;K;kGAGJ,6B;MA70JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAs1JnB,gB;K;kGAGJ,6B;MA/0JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAw1JnB,gB;K;kGAGJ,6B;MAj1JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA01JnB,gB;K;kGAGJ,6B;MAn1JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA41JnB,gB;K;kGAGJ,6B;MAr1JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAg2JnB,gB;K;kGAGJ,6B;MAz1JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAk2JnB,gB;K;kGAGJ,6B;MA31JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAo2JnB,gB;K;kGAGJ,6B;MA71JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAs2JnB,gB;K;kGAGJ,6B;MA/1JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAw2JnB,gB;K;kGAGJ,6B;MAj2JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA02JnB,gB;K;kGAGJ,6B;MAn2JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA42JnB,gB;K;kGAGJ,6B;MAr2JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAg3JnB,gB;K;kGAGJ,6B;MAz2JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAk3JnB,gB;K;kGAGJ,6B;MA32JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAo3JnB,gB;K;kGAGJ,6B;MA72JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAs3JnB,gB;K;kGAGJ,6B;MA/2JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAw3JnB,gB;K;kGAGJ,6B;MAj3JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA03JnB,gB;K;kGAGJ,6B;MAn3JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA43JnB,gB;K;kGAGJ,6B;MAr3JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAg4JnB,gB;K;kGAGJ,6B;MAz3JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAk4JnB,gB;K;kGAGJ,6B;MA33JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAo4JnB,gB;K;kGAGJ,6B;MA73JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAs4JnB,gB;K;kGAGJ,6B;MA/3JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAw4JnB,gB;K;kGAGJ,6B;MAj4JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA04JnB,gB;K;kGAGJ,6B;MAn4JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MA44JnB,gB;K;kGAGJ,6B;MAr4JiB,gB;MADb,YAAY,C;MACZ,iD;QAAa,WAAb,e;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;MAg5J

C;;QAElB,OAAO,W;O;KAnBX,C;kGAsBA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAh+UO,qBAAQ,CAg+Uf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;kGAsBA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA9+UO,qBAAQ,CA8+Uf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;kGAsBA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA5/UO,qBAAQ,CA4/Uf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;kGAsBA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1gVO,qBAAQ,CA0gVf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;kGAsBA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAxhVO,qBAAQ,CAwhVf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;kGAsBA,yB;MAAA,4F;MAAA,8D;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IAtiVO,qBAAQ,CAsiVf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,WAAjB,EAA8B,sBAAK,KAAL,CAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;4GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA5nVO,qBAAQ,CA4nVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1oVO,qBAAQ,CA0oVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAxpVO,qBAAQ,CAwpVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAtqVO,qBAAQ,CAsqVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IApsVO,qBAAQ,CAosVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAltVO,qBAAQ,CAktVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAhuVO,qBAAQ,CAguVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA9uVO,qBAAQ,CA8uVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA5vVO,qBAAQ,CA4vVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1wVO,qBAAQ,CA0wVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAxxVO,qBAAQ,CAwxVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAtyVO,qBAAQ,CAsyVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IApzVO,qBAAQ,CAozVf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAl0VO,qBAAQ,CAk0Vf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA11VO,qBAAQ,CA01Vf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA12VO,qBAAQ,CA02Vf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAl4VO,qBAAQ,CAk4Vf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA15VO,qBAAQ,CA05Vf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA16VO,qBAAQ,CA06Vf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAl8VO,qBAAQ,CAk8Vf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA19VO,qBAAQ,CA09Vf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1+VO,qBAAQ,CA0+Vf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlgWO,qBAAQ,CAkgWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1hWO,qBAAQ,CA0hWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAljWO,qBAAQ,CAkjWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1kWO,qBAAQ,CA0kWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlmWO,qBAAQ,CAkmWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1nWO,qBAAQ,CA0nWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlpWO,qBAAQ,CAkpWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1qWO,qBAAQ,CA0qWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlsWO,qBAAQ,CAksWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1tWO,qBAAQ,CA0tWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlvWO,qBAAQ,CAkvWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1wWO,qBAAQ,CA0wWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlxWO,qBAAQ,CAkxWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1yWO,qBAAQ,CA0yWf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAl0WO,qBAAQ,CAk0Wf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA11WO,qBAAQ,CA01Wf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA12WO,qBAAQ,CA02Wf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAl4WO,qBAAQ,CAk4Wf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA15WO,qBAAQ,CA05Wf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA16WO,qBAAQ,CA06Wf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAl8WO,qBAAQ,CAk8Wf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA19WO,qBAAQ,CA09Wf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1+WO,qBAAQ,CA0+Wf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlgXO,qBAAQ,CAkgXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1hXO,qBAAQ,CA0hXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAljXO,qBAAQ,CAkjXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1kXO,qBAAQ,CA0kXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlmXO,qBAAQ,CAkmXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1nXO,qBAAQ,CA0nXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlpXO,qBAAQ,CAkpXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1qXO,qBAAQ,CA0qXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlsXO,qBAAQ,CAksXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1tXO,qBAAQ,CA0tXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlvXO,qBAAQ,CAkvXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1wXO,qBAAQ,CA0wXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlxXO,qBAAQ,CAkxXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1yXO,qBAAQ,CA0yXf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAl0XO,qBAAQ,CAk0Xf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA11XO,qBAAQ,CA01Xf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA12XO,qBAAQ,CA02Xf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAl4XO,qBAAQ,CAk4Xf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA15XO,qBAAQ,CA05Xf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA16XO,qBAAQ,CA06Xf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAl8XO,qBAAQ,CAk8Xf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA19XO,qBAAQ,CA09Xf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1+XO,qBAAQ,CA0+Xf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAlgYO,qBAAQ,CAkgYf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1hYO,qBAAQ,CA0hYf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IAljYO,qBAAQ,CAkjYf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAElB,OAAO,W;O;KAnBX,C;8GAsBA,yB;MAAA,8D;MAAA,uC;QAgBqB,Q;QAHjB,IA1kYO,qBAAQ,CA0kYf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc

AQ,CAg3Vf,C;UACI,OAAO,I;QACX,kBAAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAuB,UAAK,KAAL,CAAvB,C;;QAElB,OAAO,W;O;KApBX,C;gGAuBA,yB;MAAA,8D;MAAA,uC;QAiBqB,Q;QAHjB,IA/3VO,qBAAQ,CA+T

Q;QAFtB,YAAY,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,UAAI,YAAJ,EAAI,oBAAJ,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,UAAI,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;O;KApBX,C;4GAuBA,yB;MAAA,8D;MAAA,4F;MAAA,uC;QAe0B,

,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,cAAJ,EAAI,sBAAJ,SAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KApBX,C;0GAuBA,yB;MAAA,8D;MAAA,uC;QAgB0B,UAEU,M;QAJhC,YAAY,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,OAAO,I;QACtB,kBAAkB,UAAI,YAAJ,EAAI,oBAAJ,O;QAClB

d,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KArBX,C;8FAwBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAiBoB,Q;QAHhB,IAp7XO,qBAAQ,CAo7Xf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBznnBO,W;QjB0nnBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KArBX,C;8FAwBA,yB;MAAA,gD;MAAA,gE;MAAA,oC;MAAA,gC;MAAA,gD;QAiBoB,Q;QAHhB,IAp8XO,qBAAQ,CAo8Xf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBjpnBO,W;QjBkpnBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,oBAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KArBX,C;0GAwBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA5hYO,qBAAQ,CA4hYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBzqnBO,W;QjB0qnBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA9iYO,qBAAQ,CA8iYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBrsnBO,W;QjBssnBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAhkYO,qBAAQ,CAgkYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBjunBO,W;QjBkunBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAllYO,qBAAQ,CAklYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBvwnBO,W;QjBwwnBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IApmYO,qBAAQ,CAomYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBzynBO,W;QjB0ynBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAtnYO,qBAAQ,CAsnYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB10nBO,W;QjB20nBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAxoYO,qBAAQ,CAwoYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB31nBO,W;QjB41nBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA1pYO,qBAAQ,CA0pYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB54nBO,W;QjB64nBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA5qYO,qBAAQ,CA4qYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB75nBO,W;QjB85nBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA9rYO,qBAAQ,CA8rYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBl7nBO,W;QjBm7nBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAptYO,qBAAQ,CAotYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBx8nBO,W;QjBy8nBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA1uYO,qBAAQ,CA0uYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB99nBO,W;QjB+9nBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAhwYO,qBAAQ,CAgwYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBp/nBO,W;QjBq/nBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAtxYO,qBAAQ,CAsxYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB1goBO,W;QjB2goBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA5yYO,qBAAQ,CA4yYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBhioBO,W;QjBiioBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAlzYO,qBAAQ,CAkzYf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBtpoBO,W;QjBupoBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAx0YO,qBAAQ,CAw0Yf,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB5qoBO,W;QjB6qoBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA91YO,qBAAQ,CA81YX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBlsoBO,W;QjBmsoBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAp3YO,qBAAQ,CAo3YX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBxtoBO,W;QjBytoBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA14YO,qBAAQ,CA04YX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB9uoBO,W;QjB+uoBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAh5YO,qBAAQ,CAg5YX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBpwoBO,W;QjBqwoBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAt6YO,qBAAQ,CAs6YX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB1xoBO,W;QjB2xoBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA57YO,qBAAQ,CA47YX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBhzoBO,W;QjBizoBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAl8YO,qBAAQ,CAk8YX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBt0oBO,W;QjBu0oBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAp9YO,qBAAQ,CAo9YX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB51oBO,W;QjB61oBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAt+YO,qBAAQ,CAs+YX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBh3oBO,W;QjBi3oBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAx/YO,qBAAQ,CAw/YX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBv4oBO,W;QjBw4oBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA1gZO,qBAAQ,CA0gZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB95oBO,W;QjB+5oBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA5hZO,qBAAQ,CA4hZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBn7oBO,W;QjBo7oBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAjjZO,qBAAQ,CAijZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBx8oBO,W;QjBy8oBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAtkZO,qBAAQ,CAskZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB79oBO,W;QjB89oBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA3lZO,qBAAQ,CA2lZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBl/oBO,W;QjBm/oBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAhnZO,qBAAQ,CAgnZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBvgpBO,W;QjBwgpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAznZO,qBAAQ,CAynZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB5hpBO,W;QjB6hpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA7oZO,qBAAQ,CA6oZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBjjpBO,W;QjBkjpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAlqZO,qBAAQ,CAkqZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBtkpBO,W;QjBukpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAtrZO,qBAAQ,CAsrZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB3lpBO,W;QjB4lpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA3sZO,qBAAQ,CA2sZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBhnpBO,W;QjBinpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA5tZO,qBAAQ,CA4tZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBpopBO,W;QjBqopBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAjvZO,qBAAQ,CAivZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBxppBO,W;QjBypbP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAvwZO,qBAAQ,CAuwZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB7qpBO,W;QjB8qpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAxxZO,qBAAQ,CAwxZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBlspBO,W;QjBmspBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAzyZO,qBAAQ,CAyyZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBvtpBO,W;QjBwtpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA1zZO,qBAAQ,CA0zZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB5upBO,W;QjB6upBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA30ZO,qBAAQ,CA20ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBjvpBO,W;QjBkvpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA51ZO,qBAAQ,CA41ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBtwpBO,W;QjBuwpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA72ZO,qBAAQ,CA62ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB3xpBO,W;QjB4xpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA93ZO,qBAAQ,CA83ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBhzpBO,W;QjBizpBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IA/4ZO,qBAAQ,CA+4ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBr0pBO,W;QjBs0pBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAh6ZO,qBAAQ,CAg6ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB11pBO,W;QjB21pBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAj7ZO,qBAAQ,CAi7ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB/2pBO,W;QjBg3pBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAl8ZO,qBAAQ,CAk8ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBp4pBO,W;QjBq4pBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAn9ZO,qBAAQ,CAm9ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBz5pBO,W;QjB05pBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAp+ZO,qBAAQ,CAo+ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiB96pBO,W;QjB+6pBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;4GAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAcI,IAr/ZO,qBAAQ,CAq/ZX,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,aiBn8pBO,W;QjBo8pBP,kBAAkB,O;QAClB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,gB;UACI,cAAc,UAAU

,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAnyYO,qBAAQ,CAmyYf,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACkC,kBAArB,eAAe,gBAAf,C;QAA6B,sBAAI,aAAJ,C;QAA1C,aiBz8nBO,W;QjB08nBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAhzYO,qBAAQ,CAgzYf,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACmC,kBAAtB,eAAgB,gBAAhB,C;QAA8B,sBAAI,aAAJ,C;QAA3C,aiB99nBO,W;QjB+9nBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA7zYO,qBAAQ,CA6zYf,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACoC,kBAAvB,eAAiB,gBAAjB,C;QAA+B,sBAAI,aAAJ,C;QAA5C,aiBn/nBO,W;QjBo/nBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAl1YO,qBAAQ,CAk1Yf,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACqC,kBAAxB,eAAkB,gBAAlB,C;QAAgC,sBAAI,aAAJ,C;QAA7C,aiBx/nBO,W;QjBy/nBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAv2YO,qBAAQ,CAu2Yf,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,aiB7goBO,W;QjB8goBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA53YO,qBAAQ,CA43YF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,aiBlioBO,W;QjBmioBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAj5YO,qBAAQ,CAi5YF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiBvpoBO,W;QjBwpoBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAt6YO,qBAAQ,CAs6YF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,aiB5qoBO,W;QjB6qoBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA37YO,qBAAQ,CA27YF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,aiBjsoBO,W;QjBksoBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAh9YO,qBAAQ,CAg9YF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,aiBttoBO,W;QjButoBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAr+YO,qBAAQ,CAq+YF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiB3uoBO,W;QjB4uoBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA1/YO,qBAAQ,CA0/YF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,aiBhwoBO,W;QjBiwoBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA/gZO,qBAAQ,CA+gZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,aiBtxoBO,W;QjBuxoBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAphZO,qBAAQ,CAohZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,aiB5yoBO,W;QjB6yoBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAzjZO,qBAAQ,CAyjZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiBl0oBO,W;QjBm0oBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAnlZO,qBAAQ,CAmlZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,aiB31oBO,W;QjB41oBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAxmZO,qBAAQ,CAwmZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,aiBj3oBO,W;QjBk3oBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA7nZO,qBAAQ,CA6nZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,aiBv4oBO,W;QjBw4oBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAlpZO,qBAAQ,CAkpZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,ai56oBO,W;QjB66oBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAvqZO,qBAAQ,CAuqZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,aiBj8oBO,W;QjBk8oBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA5rZO,qBAAQ,CA4rZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,aiBt9oBO,W;QjBu9oBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAjtZO,qBAAQ,CAitZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,ai3+oBO,W;QjB4+oBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA1uZO,qBAAQ,CA0uZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiBhgpBO,W;QjBigpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA/vZO,qBAAQ,CA+vZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,aiBrhpBO,W;QjBshpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IApxZO,qBAAQ,CAoxZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,ai1ipBO,W;QjB2ipBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAzyZO,qBAAQ,CAyyZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,ai5jpBO,W;QjB6jpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA9zZO,qBAAQ,CA8zZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiBlkpBO,W;QjBmkpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAn1ZO,qBAAQ,CAm1ZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,aiBvlpBO,W;QjBwlpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAx2ZO,qBAAQ,CAw2ZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,ai3mpBO,W;QjB4mpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA73ZO,qBAAQ,CA63ZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,ai7npBO,W;QjB8npBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAl5ZO,qBAAQ,CAk5ZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiBpqpBO,W;QjBqqpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAv6ZO,qBAAQ,CAu6ZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,aiBzrpBO,W;QjB0rpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA57ZO,qBAAQ,CA47ZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,ai3spBO,W;QjB4spBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAj8ZO,qBAAQ,CAi8ZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,ai7tpBO,W;QjB8tpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAt9ZO,qBAAQ,CAs9ZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiBpupBO,W;QjBqupBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA3+ZO,qBAAQ,CA2+ZF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,aizvpBO,W;QjB0vpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAhgaO,qBAAQ,CAggaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,ai3wpBO,W;QjB4wpBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAriaO,qBAAQ,CAqiaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,ai7ypBO,W;QjB8ypBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA1jaO,qBAAQ,CA0jaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiB10pBO,W;QjB20pBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA/kaO,qBAAQ,CA+kaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,aiB/1pBO,W;QjBg2pBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAplaO,qBAAQ,CAolaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,aiBn3pBO,W;QjBo3pBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAzmaO,qBAAQ,CAymaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,ai74pBO,W;QjB84pBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA5naO,qBAAQ,CA4naF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiBp6pBO,W;QjBq6pBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAjpaO,qBAAQ,CAipaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,ai/6pBO,W;QjBg7pBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAtqaO,qBAAQ,CAsqaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,ain8pBO,W;QjBo8pBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA3raO,qBAAQ,CA2raF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,aiBv9pBO,W;QjBw9pBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAhtaO,qBAAQ,CAgtaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiBj/pBO,W;QjBk/pBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAztaO,qBAAQ,CAytaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,ai5gqBO,W;QjB6gqBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAluaO,qBAAQ,CAkuaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,aiBhiqBO,W;QjBiiqBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA/uaO,qBAAQ,CA+uaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,aiBpjqBO,W;QjBqjqBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IA5vaO,qBAAQ,CA4vaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiBzkqBO,W;QjB0kqBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAzwaO,qBAAQ,CAywaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACyC,kBAA5B,eAAsB,gBAAtB,C;QAAoC,sBAAI,aAAJ,C;QAAjD,aiBjmqBO,W;QjBkmqBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAtxaO,qBAAQ,CAsxaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,aiBxnqBO,W;QjBynqBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAnyaO,qBAAQ,CAmyaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACuC,kBAA1B,eAAoB,gBAApB,C;QAAkC,sBAAI,aAAJ,C;QAA/C,aiBvoqBO,W;QjBwoqBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAlBX,C;kGAqBA,yB;MAAA,qD;MAAA,gE;MAAA,uC;QAWI,IAhzaO,qBAAQ,CAgzaF,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACwC,kBAA3B,eAAqB,gBAArB,C;QAAmC,sBAAI,aAAJ,C;QAAhD,aiB5pqBO,W;QjB6pqBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,aAAV,EAAuB,UAAK,KAAL,CAAvB,C;UACd,MA

AAA,qD;MAAA,gE;MAAA,uC;QAYI,IA9hZO,qBAAQ,CA8hZf,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACsC,kBAAzB,eAAmB,gBAAnB,C;QAAiC,sBAAI,aAAJ,C;QAA9C,aiBpuoBO,W;QjBquoBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,UAAU,KAAV,EAAiB,aAAjB,EAA8B,UAAK,KAAL,CAA9B,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAnBX,C;gHAsBA,yB;MAAA,qD;MAAA,gE;MAAA,oC;MAAA,gC;MAAA,uC;QAYI,IA5iZO,qBAAQ,CA4iZf,C;UAAe,OAAO,W;QACtB,sBAAkB,UAAK,CAAL,CAAlB,C;QACmC,kBAAtB,eAAgB,gBAAhB,C;QAA8B,sBAAI,0BAAJ,C;QAA3C,aiB1voBO,W;QjB2voBP,iBAAc,CAAd,UAAsB,gBAAtB,U;UACI,gBAAc,oBAAU,KAAV,EAAiB,0BAAjB,EAA8B,sBAAK,KAAL,EAAY,EAA9C,B,E;UACd,MAAO,WAAI,0BAAJ,C;;QAEX,OAAO,M;O;KAnBX,C;8EAsBA,yB;MAAA,gD;MAAA,gE;QAAA,sB;QAAA,sB;QAAA,sB

AAjB,EAA8B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAmwBP,gC;O;KAfJ,C;8FAkBA,yB;MAnwBA,gD;MAAA,gE;MAmwBA,gD;QAgBW,6B;;UApwBP,IA7iYO,qBAAQ,CA6iYf,C;YAAe,4BAAO,OAowBI,OApwBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAmwBlB,OAnwBkB,C;UAA5C,aiBlsnBO,W;UjBmsnBP,kBAkwB0B,O;UAjwB1B,wD;YACI,cAgwB+B,SAhwBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QA6vBP,gC;O;KAhBJ,C;8FAmBA,yB;MA7vBA,gD;MAAA,gE;MA6vBA,gD;QAgBW,6B;;UA9vBP,IA8jYO,qBAAQ,CA8jYf,C;YAAe,4BAAO,OA9vBI,OA8vBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBA6vBlB,OA7vBkB,C;UAA5C,aiB3tnBO,W;UjB4tnBP,kBA4vB0B,O;UA3vB1B,wD;YACI,cA2vB+B,SA3vBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAuuBP,gC;O;KAhBJ,C;8FAmBA,yB;MAzuBA,gD;MAAA,gE;MAyuBA,gD;QAgBW,6B;;UAvuBP,IAwlYO,qBAAQ,CAxlYf,C;YAAe,4BAAO,OAvuBI,OAuuBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAsuBlB,OAtuBkB,C;UAA5C,aiBpunBO,W;UjBqunBP,kBAquB0B,O;UApuB1B,wD;YACI,cAouB+B,SApuBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAgtBP,gC;O;KAhBJ,C;8FAmBA,yB;MAltBA,gD;MAAA,gE;MAktBA,gD;QAgBW,6B;;UAhtBP,IAqnYO,qBAAQ,CArnYf,C;YAAe,4BAAO,OAhtBI,OAgtBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBA+sBlB,OA/sBkB,C;UAA5C,aiBhsnBO,W;UjBisnBP,kBA8sB0B,O;UA7sB1B,wD;YACI,cA6sB+B,SA7sBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAyrBP,gC;O;KAhBJ,C;8FAmBA,yB;MA3rBA,gD;MAAA,gE;MA2rBA,gD;QAgBW,6B;;UAzrBP,IA0jYO,qBAAQ,CA1jYf,C;YAAe,4BAAO,OAzrBI,OAyrBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAwrBlB,OAxrBkB,C;UAA5C,aiBzqnBO,W;UjB0qnBP,kBAurB0B,O;UAtrB1B,wD;YACI,cAsrB+B,SAtrBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAkqBP,gC;O;KAhBJ,C;8FAmBA,yB;MApqBA,gD;MAAA,gE;MAoqBA,gD;QAgBW,6B;;UAlqBP,IA+hYO,qBAAQ,CA/hYf,C;YAAe,4BAAO,OAlqBI,OAkqBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAiqBlB,OAjqBkB,C;UAA5C,aiBlpnBO,W;UjBmpnBP,kBAgqB0B,O;UA/pB1B,wD;YACI,cA+pB+B,SA/pBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QA2oBP,gC;O;KAhBJ,C;8FAmBA,yB;MAjpBA,gD;MAAA,gE;MAipBA,gD;QAgBW,6B;;UA3oBP,IAoiYO,qBAAQ,CApiYf,C;YAAe,4BAAO,OA3oBI,OA2oBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBA0oBlB,OA1oBkB,C;UAA5C,aiB3onBO,W;UjB4onBP,kBAyoB0B,O;UAxoB1B,wD;YACI,cAwoB+B,SAxoBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAonBP,gC;O;KAhBJ,C;8FAmBA,yB;MA1nBA,gD;MAAA,gE;MA0nBA,gD;QAgBW,6B;;UApnBP,IAyfO,qBAAQ,CAzfR,C;YAAe,4BAAO,OApnBI,OAonBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAmnBlB,OAnnBkB,C;UAA5C,aiBrmnBO,W;UjBsmnBP,kBAknB0B,O;UAjnB1B,wD;YACI,cAinB+B,SAjnBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QA6lBP,gC;O;KAhBJ,C;8FAmBA,yB;MAnmBA,gD;MAAA,gE;MAmmBA,gD;QAgBW,6B;;UA7lBP,IA8dO,qBAAQ,CA9dR,C;YAAe,4BAAO,OA7lBI,OA6lBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBA4lBlB,OA5lBkB,C;UAA5C,aiBpknBO,W;UjBqknBP,kBA2lB0B,O;UA1lB1B,wD;YACI,cA0lB+B,SA1lBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAskBP,gC;O;KAhBJ,C;8FAmBA,yB;MA5kBA,gD;MAAA,gE;MA4kBA,gD;QAgBW,6B;;UAtkBP,IAmcO,qBAAQ,CAncR,C;YAAe,4BAAO,OAtkBI,OAskBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAqkBlB,OArkBkB,C;UAA5C,aiBnjnBO,W;UjBojnBP,kBAokB0B,O;UAnkB1B,wD;YACI,cAmkB+B,SAnkBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QA+iBP,gC;O;KAhBJ,C;8FAmBA,yB;MAriBA,gD;MAAA,gE;MAqiBA,gD;QAgBW,6B;;UA/hBP,IAwaO,qBAAQ,CAxaR,C;YAAe,4BAAO,OA/hBI,OA+hBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBA8hBlB,OA9hBkB,C;UAA5C,aiBlinBO,W;UjBminBP,kBA6hB0B,O;UA5hB1B,wD;YACI,cA4hB+B,SA5hBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAwgBP,gC;O;KAhBJ,C;8FAmBA,yB;MA9gBA,gD;MAAA,gE;MA8gBA,gD;QAgBW,6B;;UAxgBP,IAqYO,qBAAQ,CArYR,C;YAAe,4BAAO,OAxgBI,OAwgBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAugBlB,OAvgBkB,C;UAA5C,aiBjhnBO,W;UjBkhnBP,kBAsgB0B,O;UArgB1B,wD;YACI,cAqgB+B,SArgBjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAifP,gC;O;KAhBJ,C;8FAmBA,yB;MAvfA,gD;MAAA,gE;MAufA,gD;QAgBW,6B;;UAjfP,IAkXO,qBAAQ,CAlXR,C;YAAe,4BAAO,OAjfI,OAifJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAgflB,OAhfkB,C;UAA5C,aiB/gnBO,W;UjBghnBP,kBA+e0B,O;UA9e1B,wD;YACI,cA8e+B,SA9ejB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QA0dP,gC;O;KAhBJ,C;8FAmBA,yB;MAhdA,gD;MAAA,gE;MAgdA,gD;QAgBW,6B;;UA1cP,IA+VO,qBAAQ,CA/VR,C;YAAe,4BAAO,OA1cI,OA0cJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAycnB,OAzcmB,C;UAA5C,aiB9fnBO,W;UjB+fmBP,kBAwc2B,O;UAvc3B,wD;YACI,cAuc+B,SAvcjB,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAmbP,gC;O;KAhBJ,C;8FAmBA,yB;MAzaA,gD;MAAA,gE;MAyaA,gD;QAgBW,6B;;UAnaP,IA4TO,qBAAQ,CA5TR,C;YAAe,4BAAO,OAnaI,OAmaJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAkanB,OAlamB,C;UAA5C,aiB7enBO,W;UjB8emBP,kBAiaiB,O;UAhajB,wD;YACI,cAgaqB,SAhaP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QA4YP,gC;O;KAhBJ,C;8FAmBA,yB;MAlZA,gD;MAAA,gE;MAkZA,gD;QAgBW,6B;;UA5YP,IAyRO,qBAAQ,CAzRR,C;YAAe,4BAAO,OA5YI,OA4YJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBA2YnB,OA3YmB,C;UAA5C,aiB5dnBO,W;UjB6dmBP,kBA0YiB,O;UAzYjB,wD;YACI,cAyYqB,SAzYP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAqXP,gC;O;KAhBJ,C;8FAmBA,yB;MA3WA,gD;MAAA,gE;MA2WA,gD;QAgBW,6B;;UArWP,IAsPO,qBAAQ,CAtPR,C;YAAe,4BAAO,OArWI,OAqWJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAoWnB,OApWmB,C;UAA5C,aiB3cnBO,W;UjB4cmBP,kBAmWiB,O;UAlWjB,wD;YACI,cAkWqB,SAlWP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QA8UP,gC;O;KAhBJ,C;8FAmBA,yB;MApUA,gD;MAAA,gE;MAoUA,gD;QAgBW,6B;;UA9TP,IAmOO,qBAAQ,CAnOR,C;YAAe,4BAAO,OA9TI,OA8TJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBA6TnB,OA7TmB,C;UAA5C,aiBxbnBO,W;UjBybmBP,kBA4TiB,O;UA3TjB,wD;YACI,cA2TqB,SA3TP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAuSP,gC;O;KAhBJ,C;8FAmBA,yB;MA7RA,gD;MAAA,gE;MA6RA,gD;QAgBW,6B;;UAvRP,IAgMO,qBAAQ,CAhMR,C;YAAe,4BAAO,OAvRI,OAuRJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAsRnB,OAtRmB,C;UAA5C,aiBrbnBO,W;UjBsbmBP,kBAqRiB,O;UApRjB,wD;YACI,cAoRqB,SApRP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAgQP,gC;O;KAhBJ,C;8FAmBA,yB;MAtPA,gD;MAAA,gE;MAsPA,gD;QAgBW,6B;;UAhPP,IA6IO,qBAAQ,CA7IR,C;YAAe,4BAAO,OAhPI,OAgPJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBA+OnB,OA/OmB,C;UAA5C,aiBlanBO,W;UjBmanBP,kBA8OiB,O;UA7OjB,wD;YACI,cA6OqB,SA7OP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAyNP,gC;O;KAhBJ,C;8FAmBA,yB;MA/MA,gD;MAAA,gE;MA+MA,gD;QAgBW,6B;;UAzMP,IA0GO,qBAAQ,CA1GR,C;YAAe,4BAAO,OAzMI,OAyMJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAwMnB,OAxMmB,C;UAA5C,aiB5ZnBO,W;UjB6ZmBP,kBAuMiB,O;UAtMjB,wD;YACI,cAsMqB,SAtMP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAkLP,gC;O;KAhBJ,C;8FAmBA,yB;MAxKA,gD;MAAA,gE;MAwKA,gD;QAgBW,6B;;UAlKP,IAuEO,qBAAQ,CAvER,C;YAAe,4BAAO,OAlKI,OAkKJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAiKnB,OAjKmB,C;UAA5C,aiBnYnBO,W;UjBoYmBP,kBAgKiB,O;UA/JjB,wD;YACI,cA+JqB,SA/JP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QA2IP,gC;O;KAhBJ,C;8FAmBA,yB;MAjIA,gD;MAAA,gE;MAiIA,gD;QAgBW,6B;;UA3HP,IAoBO,qBAAQ,CApBR,C;YAAe,4BAAO,OA3HI,OA2HJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBA0HnB,OA1HmB,C;UAA5C,aiB1WnBO,W;UjB2WmBP,kBAyHiB,O;UAxHjB,wD;YACI,cAwHqB,SAxHP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAoGP,gC;O;KAhBJ,C;8FAmBA,yB;MA1FA,gD;MAAA,gE;MA0FA,gD;QAgBW,6B;;UApFP,IAaO,qBAAQ,CAbR,C;YAAe,4BAAO,OApFI,OAoFJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAmFnB,OAnFmB,C;UAA5C,aiBjVnBO,W;UjBkVmBP,kBAkFiB,O;UAjFjB,wD;YACI,cAiFqB,SAjFP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QA6DP,gC;O;KAhBJ,C;8FAmBA,yB;MAnDA,gD;MAAA,gE;MAmDA,gD;QAgBW,6B;;UA7CP,IAMO,qBAAQ,CANR,C;YAAe,4BAAO,OA7CI,OA6CJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBA4CnB,OA5CmB,C;UAA5C,aiBxUnBO,W;UjByUmBP,kBA2CiB,O;UA1CjB,wD;YACI,cA0CqB,SA1CP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAsBP,gC;O;KAhBJ,C;8FAmBA,yB;MA5AA,gD;MAAA,gE;MA4AA,gD;QAgBW,6B;;UAtBP,IAwBO,qBAAQ,CAxBR,C;YAAe,4BAAO,OAtBI,OAsBJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAqBnB,OArBmB,C;UAA5C,aiBnTnBO,W;UjBoTmBP,kBAoBiB,O;UAnBjB,wD;YACI,cAmBqB,SAnBP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAeP,gC;O;KAhBJ,C;8FAmBA,yB;MArBA,gD;MAAA,gE;MAqBA,gD;QAgBW,6B;;UAfP,IAiBO,qBAAQ,CAjBR,C;YAAe,4BAAO,OAfI,OAeJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAcnB,OAdmB,C;UAA5C,aiB9SnBO,W;UjB+SmBP,kBAaiB,O;UAZjB,wD;YACI,cAYqB,SAZP,CAAU,KAAV,EAAiB,WAAjB,EAA8B,EAA9B,UAAK,KAAL,CAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAQP,gC;O;KAhBJ,C;8FAmBA,yB;MAdA,gD;MAAA,gE;MAcA,gD;QAgBW,6B;;UARP,IAUO,qBAAQ,CAVR,C;YAAe,4BAAO,OARI,OAQJ,C;YAAP,8B;;UACqB,kBAAvB,eAAa,mBAAO,CAAP,IAAb,C;UAA+B,sBAOnB,OAPmB,C;UAA5C,aiBzRnBO,W;UjB0RmBP,kBAMiB,O;UALjB,wD;YACI,cAKqB,SAL

AAhB,M;QACI,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;K;8FAGX,+B;MAOoB,Q;MADhB,UAAkB,G;MAClB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SA

Q;QADhB,Y;QACA,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,cAAO,SAAS,OAAT,CAAP,C;;QAEJ,OAAO,G;O;KAbX,C;mFAgBA,yB;MAAA,SASoB,gB;MATpB,sC;QAUoB,Q;QADhB,Y;QACA,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,cAAO,SAAS,OAAT,CAAP,C;;QAEJ,OAAO,G;O;KAbX,C;mFAgBA,yB;MAAA,SASoB,oC;MAAA,gC;MAAA,sC;QAUoB,Q;QADhB,Y;QACA,wBAAgB,SAAhB,gB;UAAgB,cAAhB,B,UAAgB,SAAhB,B,O;UACI,cAAO,SAAS,oBAAT,CAAP,C;;QAEJ,OAAO,G;O;KAbX,C;mFAgBA,yB;MAAA,6B,6B;MAAA,sC;QAUoB,Q;QADhB,UAAgB,7pBAAhB,C,cAAgB,7pBAAhB,B,C;QAAgB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,cAAO,SAAS,O

qqB/B,CmB7qqB+B,CAAX,C;QnB8qqBrC,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,MmBl/qBmD,
enBk/qBnD,GmBl/qB8D,KAAK,KnBk/qB5D,SAAS,OAAT,CmBl/qBuE,KAAX,CAAhB,C;;QnBo/qBvD,OAAO,
G;O;KAdX,C;kFAiBA,yB;MmB/rqBA,+B;MnB+rqBA,sC;QAWoB,Q;QADhB,UmB9rqBqC,eAAW,oBnB8rqB/B,
CmB9rqB+B,CAAX,C;QnB+rqBrC,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,MmBngrBmD,enBmg
rBnD,GmBngrB8D,KAAK,KnBmgrB5D,SAAS,OAAT,CmBngrBuE,KAAX,CAAhB,C;;QnBqgrBvD,OAAO,G;O;
KAdX,C;mFAiBA,yB;MAAA,oC;MAAA,gC;MmBhtqBA,+B;MnBgtqBA,sC;QAWoB,Q;QADhB,UmB/sqBqC,e
AAW,oBnB+sqB/B,CmB/sqB+B,CAAX,C;QnBgtqBrC,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O
;UACI,MmBphrBmD,enBohrBnD,GmBphrB8D,KAAK,KnBohrB5D,SAAS,oBAAT,CmBphrBuE,KAAX,CAAhB,
C;;QnBshrBvD,OAAO,G;O;KAdX,C;IAiBA,mC;MAIoB,UAMT,M;MANP,wBAAgB,SAAhB,gB;QAAgB,cAAA,
SAAhB,M;QACI,IAAI,eAAJ,C;UACI,MAAM,gCAAyB,2BAAwB,SAAxB,MAAzB,C;;;MAId,OAAO,0D;K;wFA
GX,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAUoB,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,SAAh
B,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,MAA
O,WAAI,OAAJ,C;;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;O;KAjBX,C;0FAoBA,yB;MAAA,+D;MAAA,6
B;MAAA,uC;QAUoB,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAh
B,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;;QAGf,
OAAO,cAAK,KAAL,EAAY,MAAZ,C;O;KAjBX,C;0FAoBA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAUoB,Q;Q
AFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,O
AAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;;QAGf,OAAO,cAAK,KAAL,EAA
Y,MAAZ,C;O;KAjBX,C;0FAoBA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAUoB,Q;QAFhB,YAAY,gB;QACZ,a
AAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,KAA
M,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;O;KAjBX,C;0
FAoBA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAUoB,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,S
AAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,
MAAO,WAAI,OAAJ,C;;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;O;KAjBX,C;0FAoBA,yB;MAAA,+D;MA
AA,6B;MAAA,uC;QAUoB,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAA,S
AAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;;Q
AGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;O;KAjBX,C;0FAoBA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAUoB
,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAA
U,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;;QAGf,OAAO,cAAK,KAAL,E
AAY,MAAZ,C;O;KAjBX,C;0FAoBA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAUoB,Q;QAFhB,YAAY,gB;QAC
Z,aAAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,K
AAM,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;O;KAjBX,
C;0FAoBA,yB;MAAA,+D;MAAA,oC;MAAA,gC;MAAA,6B;MAAA,uC;QAUoB,Q;QAFhB,YAAY,gB;QACZ,a
AAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UACI,IAAI,UAAU,oBAAV,CAAJ,C;Y
ACI,KAAM,WAAI,oBAAJ,C;;YAEN,MAAO,WAAI,oBAAJ,C;;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;O;
KAjBX,C;IAoBA,+B;MAkGI,WkB3orBO,MAAO,KlB2orBG,gBkB3orBH,ElBgjrBH,KA2FkB,OkB3orBf,C;MlB4
orBd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA9FqB,GA8FP,UAAK,C
AAL,CA9FO,EAAnB,KA8FqB,CAAM,CAAN,CA9FF,CA8FrB,C;;MA9FT,OAgGO,I;K;IA7FX,iC;MAwGI,WkB
3prBO,MAAO,KlB2prBG,gBkB3prBH,ElB0jrBH,KAiGkB,OkB3prBf,C;MlB4prBd,WAAW,iBAAa,IAAb,C;MA
CX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WApGqB,GAoGP,UAAK,CAAL,CApGO,EAAnB,KAoGqB,C
AAM,CAAN,CApGF,CAoGrB,C;;MApGT,OAsGO,I;K;IAnGX,iC;MA8GI,WkB3qrBO,MAAO,KlB2qrBG,gBkB3
qrBH,ElBokrBH,KAuGkB,OkB3qrBf,C;MlB4qrBd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAl
B,M;QACI,IAAK,WA1GqB,GA0GP,UAAK,CAAL,CA1GO,EAAnB,KA0GqB,CAAM,CAAN,CA1GF,CA0GrB,
C;;MA1GT,OA4GO,I;K;IAzGX,iC;MAoHI,WkB3rrBO,MAAO,KlB2rrBG,gBkB3rrBH,ElB8krBH,KA6GkB,OkB
3rrBf,C;MlB4rrBd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAhHqB,GAg
HP,UAAK,CAAL,CAhHO,EAAnB,KAgHqB,CAAM,CAAN,CAhHF,CAgHrB,C;;MAhHT,OAkHO,I;K;IA/GX,iC
;MA0HI,WkB3srBO,MAAO,KlB2srBG,gBkB3srBH,ElBwlrBH,KAmHkB,OkB3srBf,C;MlB4srBd,WAAW,iBAAa
,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAtHqB,GAsHP,UAAK,CAAL,CAtHO,EAAnB,

KAsHqB,CAAM,CAAN,CAtHF,CAsHrB,C;;MAtHT,OAwHO,I;K;IArHX,iC;MAgII,WkB3trBO,MAAO,KlB2trBG,gBkB3trBH,ElBkmrBH,KAyHkB,OkB3trBf,C;MlB4trBd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA5HqB,GA4HP,UAAK,CAAL,CA5HO,EAAnB,KA4HqB,CAAM,CAAN,CA5HF,CA4HrB,C;;MA5HT,OA8HO,I;K;IA3HX,iC;MAsII,WkB3urBO,MAAO,KlB2urBG,gBkB3urBH,ElB4mrBH,KA+HkB,OkB3urBf,C;MlB4urBd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAlIqB,GAkIP,UAAK,CAAL,CAlIO,EAAnB,KAkIqB,CAAM,CAAN,CAlIF,CAkIrB,C;;MAlIT,OAoIO,I;K;IAjIX,iC;MA4II,WkB3vrBO,MAAO,KlB2vrBG,gBkB3vrBH,ElBsnrBH,KAqIkB,OkB3vrBf,C;MlB4vrBd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAxIqB,GAwIP,UAAK,CAAL,CAxIO,EAAnB,KAwIqB,CAAM,CAAN,CAxIF,CAwIrB,C;;MAxIT,OA0IO,I;K;IAvIX,iC;MAkJI,WkB3wrBO,MAAO,KlB2wrBG,gBkB3wrBH,ElBgorBH,KA2IkB,OkB3wrBf,C;MlB4wrBd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA9IqB,GA8IP,UAAK,CAAL,CA9IO,EAAnB,KA8IqB,CAAM,CAAN,CA9IF,CA8IrB,C;;MA9IT,OAgJO,I;K;8EA7IX,yB;MAAA,gE;MkBzorBA,iB;MlByorBA,8C;QAAI,WkB3orBO,MAAO,KlB2orBG,gBkB3orBH,ElB2orBS,KAAM,OkB3orBf,C;QlB4orBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;8EAAA,yB;MAAA,gE;MkBzprBA,iB;MlByprBA,8C;QAAI,WkB3prBO,MAAO,KlB2prBG,gBkB3prBH,ElB2prBS,KAAM,OkB3prBf,C;QlB4prBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzqrBA,iB;MlByqrBA,8C;QAAI,WkB3qrBO,MAAO,KlB2qrBG,gBkB3qrBH,ElB2qrBS,KAAM,OkB3qrBf,C;QlB4qrBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;8EAAA,yB;MAAA,gE;MkBzrrBA,iB;MlByrrBA,8C;QAAI,WkB3rrBO,MAAO,KlB2rrBG,gBkB3rrBH,ElB2rrBS,KAAM,OkB3rrBf,C;QlB4rrBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzsrBA,iB;MlBysrBA,8C;QAAI,WkB3srBO,MAAO,KlB2srBG,gBkB3srBH,ElB2srBS,KAAM,OkB3srBf,C;QlB4srBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBztrBA,iB;MlBytrBA,8C;QAAI,WkB3trBO,MAAO,KlB2trBG,gBkB3trBH,ElB2trBS,KAAM,OkB3trBf,C;QlB4trBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzurBA,iB;MlByurBA,8C;QAAI,WkB3urBO,MAAO,KlB2urBG,gBkB3urBH,ElB2urBS,KAAM,OkB3urBf,C;QlB4urBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzvrBA,iB;MlByvrBA,8C;QAAI,WkB3vrBO,MAAO,KlB2vrBG,gBkB3vrBH,ElB2vrBS,KAAM,OkB3vrBf,C;QlB4vrBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzwrBA,iB;MlBywrBA,8C;QAAI,WkB3wrBO,MAAO,KlB2wrBG,gBkB3wrBH,ElB2wrBS,KAAM,OkB3wrBf,C;QlB4wrBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzxrBA,iB;MlByxrBA,8C;QAAI,WkB3xrBO,MAAO,KlB2xrBG,gBkB3xrBH,ElB2xrBS,KAAM,OkB3xrBf,C;QlB4xrBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzyrBA,iB;MlByyrBA,8C;QAAI,WkB3yrBO,MAAO,KlB2yrBG,gBkB3yrBH,ElB2yrBS,KAAM,OkB3yrBf,C;QlB4yrBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzzrBA,iB;MlByzrBA,8C;QAAI,WkB3zrBO,MAAO,KlB2zrBG,gBkB3zrBH,ElB2zrBS,KAAM,OkB3zrBf,C;QlB4zrBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz0rBA,iB;MlBy0rBA,8C;QAAI,WkB30rBO,MAAO,KlB20rBG,gBkB30rBH,ElB20rBS,KAAM,OkB30rBf,C;QlB40rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz1rBA,iB;MlBy1rBA,8C;QAAI,WkB31rBO,MAAO,KlB21rBG,gBkB31rBH,ElB21rBS,KAAM,OkB31rBf,C;QlB41rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz2rBA,iB;MlBy2rBA,8C;QAAI,WkB32rBO,MAAO,KlB22rBG,gBkB32rBH,ElB22rBS,KAAM,OkB32rBf,C;QlB42rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz3rBA,iB;MlBy3rBA,8C;QAAI,WkB33rBO,MAAO,KlB23rBG,gBkB33rBH,ElB23rBS,KAAM,OkB33rBf,C;QlB43rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz4rBA,iB;MlBy4rBA,8C;QAAI,WkB34rBO,MAAO,KlB24rBG,gBkB34rBH,ElB24rBS,KAAM,OkB34rBf,C;QlB44rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz5rBA,iB;MlBy5rBA,8C;QAAI,WkB35rBO,MAAO,KlB25rBG,gBkB35rBH,ElB25rBS,KAAM,OkB35rBf,C;QlB45rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz6rBA,iB;MlBy6rBA,8C;QAAI,WkB36rBO,MAAO,KlB26rBG,gBkB36rBH,ElB26rBS,KAAM,OkB36rBf,C;QlB46rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz7rBA,iB;MlBy7rBA,8C;QAAI,WkB37rBO,MAAO,KlB27rBG,gBkB37rBH,ElB27rBS,KAAM,OkB37rBf,C;QlB47rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz8rBA,iB;MlBy8rBA,8C;QAAI,WkB38rBO,MAAO,KlB28rBG,gBkB38rBH,ElB28rBS,KAAM,OkB38rBf,C;QlB48rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz9rBA,iB;MlBy9rBA,8C;QAAI,WkB39rBO,MAAO,KlB29rBG,gBkB39rBH,ElB29rBS,KAAM,OkB39rBf,C;QlB49rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz+rBA,iB;MlBy+rBA,8C;QAAI,WkB3+rBO,MAAO,KlB2+rBG,gBkB3+rBH,ElB2+rBS,KAAM,OkB3+rBf,C;QlB4+rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz/rBA,iB;MlBy/rBA,8C;QAAI,WkB3/rBO,MAAO,KlB2/rBG,gBkB3/rBH,ElB2/rBS,KAAM,OkB3/rBf,C;QlB4/rBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzgsBA,iB;MlBygsBA,8C;QAAI,WkB3gsBO,MAAO,KlB2gsBG,gBkB3gsBH,ElB2gsBS,KAAM,OkB3gsBf,C;QlB4gsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzhsBA,iB;MlBygsBA,8C;QAAI,WkB3hsBO,MAAO,KlB2hsBG,gBkB3hsBH,ElB2hsBS,KAAM,OkB3hsBf,C;QlB4hsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzisBA,iB;MlBygsBA,8C;QAAI,WkB3isBO,MAAO,KlB2isBG,gBkB3isBH,ElB2isBS,KAAM,OkB3isBf,C;QlB4isBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzjsBA,iB;MlBygsBA,8C;QAAI,WkB3jsBO,MAAO,KlB2jsBG,gBkB3jsBH,ElB2jsBS,KAAM,OkB3jsBf,C;QlB4jsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzksBA,iB;MlBygsBA,8C;QAAI,WkB3ksBO,MAAO,KlB2ksBG,gBkB3ksBH,ElB2ksBS,KAAM,OkB3ksBf,C;QlB4ksBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzlsBA,iB;MlBygsBA,8C;QAAI,WkB3lsBO,MAAO,KlB2lsBG,gBkB3lsBH,ElB2lsBS,KAAM,OkB3lsBf,C;QlB4lsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzmsBA,iB;MlBygsBA,8C;QAAI,WkB3msBO,MAAO,KlB2msBG,gBkB3msBH,ElB2msBS,KAAM,OkB3msBf,C;QlB4msBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBznsBA,iB;MlBygsBA,8C;QAAI,WkB3nsBO,MAAO,KlB2nsBG,gBkB3nsBH,ElB2nsBS,KAAM,OkB3nsBf,C;QlB4nsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzosBA,iB;MlBygsBA,8C;QAAI,WkB3osBO,MAAO,KlB2osBG,gBkB3osBH,ElB2osBS,KAAM,OkB3osBf,C;QlB4osBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzpsBA,iB;MlBygsBA,8C;QAAI,WkB3psBO,MAAO,KlB2psBG,gBkB3psBH,ElB2psBS,KAAM,OkB3psBf,C;QlB4psBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzqsBA,iB;MlBygsBA,8C;QAAI,WkB3qsBO,MAAO,KlB2qsBG,gBkB3qsBH,ElB2qsBS,KAAM,OkB3qsBf,C;QlB4qsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzrsBA,iB;MlBygsBA,8C;QAAI,WkB3rsBO,MAAO,KlB2rsBG,gBkB3rsBH,ElB2rsBS,KAAM,OkB3rsBf,C;QlB4rsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzssBA,iB;MlBygsBA,8C;QAAI,WkB3ssBO,MAAO,KlB2ssBG,gBkB3ssBH,ElB2ssBS,KAAM,OkB3ssBf,C;QlB4ssBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBztsBA,iB;MlBygsBA,8C;QAAI,WkB3tsBO,MAAO,KlB2tsBG,gBkB3tsBH,ElB2tsBS,KAAM,OkB3tsBf,C;QlB4tsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzusBA,iB;MlBygsBA,8C;QAAI,WkB3usBO,MAAO,KlB2usBG,gBkB3usBH,ElB2usBS,KAAM,OkB3usBf,C;QlB4usBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzvsBA,iB;MlBygsBA,8C;QAAI,WkB3vsBO,MAAO,KlB2vsBG,gBkB3vsBH,ElB2vsBS,KAAM,OkB3vsBf,C;QlB4vsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzwsBA,iB;MlBygsBA,8C;QAAI,WkB3wsBO,MAAO,KlB2wsBG,gBkB3wsBH,ElB2wsBS,KAAM,OkB3wsBf,C;QlB4wsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzxsBA,iB;MlBygsBA,8C;QAAI,WkB3xsBO,MAAO,KlB2xsBG,gBkB3xsBH,ElB2xsBS,KAAM,OkB3xsBf,C;QlB4xsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzysBA,iB;MlBygsBA,8C;QAAI,WkB3ysBO,MAAO,KlB2ysBG,gBkB3ysBH,ElB2ysBS,KAAM,OkB3ysBf,C;QlB4ysBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzzsBA,iB;MlBygsBA,8C;QAAI,WkB3zsBO,MAAO,KlB2zsBG,gBkB3zsBH,ElB2zsBS,KAAM,OkB3zsBf,C;QlB4zsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz0sBA,iB;MlBygsBA,8C;QAAI,WkB30sBO,MAAO,KlB20sBG,gBkB30sBH,ElB20sBS,KAAM,OkB30sBf,C;QlB40sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz1sBA,iB;MlBygsBA,8C;QAAI,WkB31sBO,MAAO,KlB21sBG,gBkB31sBH,ElB21sBS,KAAM,OkB31sBf,C;QlB41sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz2sBA,iB;MlBygsBA,8C;QAAI,WkB32sBO,MAAO,KlB22sBG,gBkB32sBH,ElB22sBS,KAAM,OkB32sBf,C;QlB42sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz3sBA,iB;MlBygsBA,8C;QAAI,WkB33sBO,MAAO,KlB23sBG,gBkB33sBH,ElB23sBS,KAAM,OkB33sBf,C;QlB43sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz4sBA,iB;MlBygsBA,8C;QAAI,WkB33sBO,MAAO,KlB23sBG,gBkB33sBH,ElB23sBS,KAAM,OkB33sBf,C;QlB43sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz5sBA,iB;MlBygsBA,8C;QAAI,WkB35sBO,MAAO,KlB25sBG,gBkB35sBH,ElB25sBS,KAAM,OkB35sBf,C;QlB45sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz6sBA,iB;MlBygsBA,8C;QAAI,WkB35sBO,MAAO,KlB25sBG,gBkB35sBH,ElB25sBS,KAAM,OkB35sBf,C;QlB45sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz7sBA,iB;MlBygsBA,8C;QAAI,WkB35sBO,MAAO,KlB25sBG,gBkB35sBH,ElB25sBS,KAAM,OkB35sBf,C;QlB45sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz8sBA,iB;MlBygsBA,8C;QAAI,WkB35sBO,MAAO,KlB25sBG,gBkB35sBH,ElB25sBS,KAAM,OkB35sBf,C;QlB45sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz9sBA,iB;MlBygsBA,8C;QAAI,WkB35sBO,MAAO,KlB25sBG,gBkB35sBH,ElB25sBS,KAAM,OkB35sBf,C;QlB45sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz+sBA,iB;MlBygsBA,8C;QAAI,WkB35sBO,MAAO,KlB25sBG,gBkB35sBH,ElB25sBS,KAAM,OkB35sBf,C;QlB45sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBz/sBA,iB;MlBygsBA,8C;QAAI,WkB35sBO,MAAO,KlB25sBG,gBkB35sBH,ElB25sBS,KAAM,OkB35sBf,C;QlB45sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzgtBA,iB;MlBygsBA,8C;QAAI,WkB35sBO,MAAO,KlB25sBG,gBkB35sBH,ElB25sBS,KAAM,OkB35sBf,C;QlB45sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzhtBA,iB;MlBygsBA,8C;QAAI,WkB35sBO,MAAO,KlB25sBG,gBkB35sBH,ElB25sBS,KAAM,OkB35sBf,C;QlB45sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;MkBzitBA,iB;MlBygsBA,8C;QAAI,WkB35sBO,MAAO,KlB25sBG,gBkB35sBH,ElB25sBS,KAAM,OkB35sBf,C;QlB45sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAK,EAAnB,MAAmB,CAAM,CAAN,CAAM,CAAnB,CAAmB,C;;QAAnB,OAAA,O,I;O;KAAA,X,C;+EAAA,yB;MAAA,gE;

B;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhHqB,GAgHP,UAAK,UAAL,EAAK,kBAAL,SAh
HO,EAgHI,OAhHJ,CAgHrB,C;;MAhHT,OAkHO,I;K;IA/GX,kC;MA6HoB,gB;MAHhB,gBAAgB,gB;MAChB,W
AAW,iBkB56rBJ,MAAO,KlB46rBsB,wBApHzB,KAoHyB,EAAwB,EAAxB,CkB56rBtB,ElB46rBmD,SkB56rBn
D,ClB46rBH,C;MACX,QAAQ,C;MACQ,OAtHL,KAsHK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAA
I,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxHqB,GAwHP,UAAK,UAAL,EAAK,kBAAL,SAxHO,EAwHI,O
AxHJ,CAwHrB,C;;MAxHT,OA0HO,I;K;IAvHX,kC;MAqIoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB97r
BJ,MAAO,KlB87rBsB,wBA5HzB,KA4HyB,EAAwB,EAAxB,CkB97rBtB,ElB87rBmD,SkB97rBnD,ClB87rBH,C;
MACX,QAAQ,C;MACQ,OA9HL,KA8HK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT
,C;UAAoB,K;QACpB,IAAK,WAhIqB,GAgIP,UAAK,UAAL,EAAK,kBAAL,SAhIO,EAgII,OAhIJ,CAgIrB,C;;MA
hIT,OAkIO,I;K;IA/HX,kC;MA6IoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBh9rBJ,MAAO,KlBg9rBsB,w
BApIzB,KAoIyB,EAAwB,EAAxB,CkBh9rBtB,ElBg9rBmD,SkBh9rBnD,ClBg9rBH,C;MACX,QAAQ,C;MACQ,
OAtIL,KAsIK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAxIqB,GAwIP,UAAK,UAAL,EAAK,kBAAL,SAxIO,EAwII,OAxIJ,CAwIrB,C;;MAxIT,OA0IO,I;K;IAvIX,kC;
MAqJoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBl+rBJ,MAAO,KlBk+rBsB,wBA5IzB,KA4IyB,EAAwB,EA
AxB,CkBl+rBtB,ElBk+rBmD,SkBl+rBnD,ClBk+rBH,C;MACX,QAAQ,C;MACQ,OA9IL,KA8IK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhJqB,GAgJP,UAAK,
UAAL,EAAK,kBAAL,SAhJO,EAgJI,OAhJJ,CAgJrB,C;;MAhJT,OAkJO,I;K;IA/IX,kC;MA6JoB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBp/rBJ,MAAO,KlBo/rBsB,wBApJzB,KAoJyB,EAAwB,EAAxB,CkBp/rBtB,ElBo/rBmD,
SkBp/rBnD,ClBo/rBH,C;MACX,QAAQ,C;MACQ,OAtJL,KAsJK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxJqB,GAwJP,UAAK,UAAL,EAAK,kBAAL,SAxJO,EAwJI,O
AxJJ,CAwJrB,C;;MAxJT,OA0JO,I;K;IAvJX,kC;MAqKoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBtgsBJ,MAAO,
KlBsgsBsB,wBA5JzB,KA4JyB,EAAwB,EAAxB,CkBtgsBtB,ElBsgsBmD,SkBtgsBnD,ClBsgsBH,C;MACX,QAAQ,C;MACQ,
OA9JL,KA8JK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAhKqB,GAgKP,UAAK,UAAL,EAAK,kBAAL,SAhKO,EAgKI,OAhKJ,CAgKrB,C;;MAhKT,OAkKO,I;K;IA/JX,kC;
MA6KoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBxhsBJ,MAAO,KlBwhsBsB,wBApKzB,KAoKyB,EAAwB,EA
AxB,CkBxhsBtB,ElBwhsBmD,SkBxhsBnD,ClBwhsBH,C;MACX,QAAQ,C;MACQ,OAtKL,KAsKK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxKqB,GAwKP,UAAK,
UAAL,EAAK,kBAAL,SAxKO,EAwKI,OAxKJ,CAwKrB,C;;MAxKT,OA0KO,I;K;IAvKX,kC;MAqLoB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkB1isBJ,MAAO,KlB0isBsB,wBA5KzB,KA4KyB,EAAwB,EAAxB,CkB1isBtB,ElB0isBmD,
SkB1isBnD,ClB0isBH,C;MACX,QAAQ,C;MACQ,OA9KL,KA8KK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhLqB,GAgLP,UAAK,UAAL,EAAK,kBAAL,SAhLO,EAgLI,O
AhLJ,CAgLrB,C;;MAhLT,OAkLO,I;K;IA/KX,kC;MA6MoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB5jsBJ,MAAO,
KlB4jsBsB,wBApLzB,KAoLyB,EAAwB,EAAxB,CkB5jsBtB,ElB4jsBmD,SkB5jsBnD,ClB4jsBH,C;MACX,QAAQ,C;MACQ,
OAtLL,KAsLK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAxLqB,GAwLP,UAAK,UAAL,EAAK,kBAAL,SAxLO,EAwLI,OAxLJ,CAwLrB,C;;MAxLT,OA0LO,I;K;IAvLX,kC;
MAqMoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB5ksBJ,MAAO,KlB4ksBsB,wBA5LzB,KA4LyB,EAAwB,EA
AxB,CkB5ksBtB,ElB4ksBmD,SkB5ksBnD,ClB4ksBH,C;MACX,QAAQ,C;MACQ,OA9LL,KA8LK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhMqB,GAgMP,UAAK,
UAAL,EAAK,kBAAL,SAhMO,EAgMI,OAhMJ,CAgMrB,C;;MAhMT,OAkMO,I;K;IA/LX,kC;MA6NoB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkB9lsBJ,MAAO,KlB8lsBsB,wBApMzB,KAoMyB,EAAwB,EAAxB,CkB9lsBtB,ElB8lsBmD,
SkB9lsBnD,ClB8lsBH,C;MACX,QAAQ,C;MACQ,OAtML,KAsMK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxMqB,GAwMP,UAAK,UAAL,EAAK,kBAAL,SAxMO,EAwMI,O
AxMJ,CAwMrB,C;;MAxMT,OA0MO,I;K;IAvMX,kC;MAqNoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBhnsBJ,MAAO,
KlBgnsBsB,wBA5MzB,KA4MyB,EAAwB,EAAxB,CkBhnsBtB,ElBgnsBmD,SkBhnsBnD,ClBgnsBH,C;MACX,QAAQ,C;MACQ,
OA9ML,KA8MK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAhNqB,GAgNP,UAAK,UAAL,EAAK,kBAAL,SAhNO,EAgNI,OAhNJ,CAgNrB,C;;MAhNT,OAkNO,I;K;IA/MX,kC;
MA6OoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBposBJ,MAAO,KlBoosBsB,wBApNzB,KAoNyB,EAAwB,EA
AxB,CkBposBtB,ElBoosBmD,SkBposBnD,ClBoosBH,C;MACX,QAAQ,C;MACQ,OAtNL,KAsNK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxNqB,GAwNP,UAAK,
UAAL,EAAK,kBAAL,SAxNO,EAwNI,OAxNJ,CAwNrB,C;;MAxNT,OA0NO,I;K;IAvNX,kC;MAqOoB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBtpsBJ,MAAO,KlBspsBsB,wBA5NzB,KA4NyB,EAAwB,EAAxB,CkBtpsBtB,ElBspsBmD,
SkBtpsBnD,ClBspsBH,C;MACX,QAAQ,C;MACQ,OA9NL,KA8NK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhOqB,GAgOP,UAAK,UAAL,EAAK,kBAAL,SAhOO,EAgOI,O
AhOJ,CAgOrB,C;;MAhOT,OAkOO,I;K;IA/NX,kC;MA6PoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBxqsBJ,MAAO,
KlBwqsBsB,wBApOzB,KAoOyB,EAAwB,EAAxB,CkBxqsBtB,ElBwqsBmD,SkBxqsBnD,ClBwqsBH,C;MACX,QAAQ,C;MACQ,
OAtOL,KAsOK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAxOqB,GAwOP,UAAK,UAAL,EAAK,kBAAL,SAxOO,EAwOI,OAxOJ,CAwOrB,C;;MAxOT,OA0OO,I;K;IAvOX,kC;
MAqPoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB1rsBJ,MAAO,KlB0rsBsB,wBA5OzB,KA4OyB,EAAwB,EA
AxB,CkB1rsBtB,ElB0rsBmD,SkB1rsBnD,ClB0rsBH,C;MACX,QAAQ,C;MACQ,OA9OL,KA8OK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhPqB,GAgPP,UAAK,
UAAL,EAAK,kBAAL,SAhPO,EAgPI,OAhPJ,CAgPrB,C;;MAhPT,OAkPO,I;K;IA/OX,kC;MA6QoB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkB5tsBJ,MAAO,KlB4tsBsB,wBApPzB,KAoPyB,EAAwB,EAAxB,CkB5tsBtB,ElB4tsBmD,
SkB5tsBnD,ClB4tsBH,C;MACX,QAAQ,C;MACQ,OAtPL,KAsPK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxPqB,GAwPP,UAAK,UAAL,EAAK,kBAAL,SAxPO,EAwPI,O
AxPJ,CAwPrB,C;;MAxPT,OA0PO,I;K;IAvPX,kC;MAqQoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB9vsBJ,MAAO,
KlB8vsBsB,wBA5PzB,KA4PyB,EAAwB,EAAxB,CkB9vsBtB,ElB8vsBmD,SkB9vsBnD,ClB8vsBH,C;MACX,QAAQ,C;MACQ,
OA9PL,KA8PK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAhQqB,GAgQP,UAAK,UAAL,EAAK,kBAAL,SAhQO,EAgQI,OAhQJ,CAgQrB,C;;MAhQT,OAkQO,I;K;IA/PX,kC;
MA6RoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBhysBJ,MAAO,KlBgysBsB,wBApQzB,KAoQyB,EAAwB,EA
AxB,CkBhysBtB,ElBgysBmD,SkBhysBnD,ClBgysBH,C;MACX,QAAQ,C;MACQ,OAtQL,KAsQK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxQqB,GAwQP,UAAK,
UAAL,EAAK,kBAAL,SAxQO,EAwQI,OAxQJ,CAwQrB,C;;MAxQT,OA0QO,I;K;IAvQX,kC;MAqRoB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBl0sBJ,MAAO,KlBk0sBsB,wBA5QzB,KA4QyB,EAAwB,EAAxB,CkBl0sBtB,ElBk0sBmD,
SkBl0sBnD,ClBk0sBH,C;MACX,QAAQ,C;MACQ,OA9QL,KA8QK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhRqB,GAgRP,UAAK,UAAL,EAAK,kBAAL,SAhRO,EAgRI,O
AhRJ,CAgRrB,C;;MAhRT,OAkRO,I;K;IA/QX,kC;MA6SoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBp1sBJ,MAAO,
KlBo1sBsB,wBApRzB,KAoRyB,EAAwB,EAAxB,CkBp1sBtB,ElBo1sBmD,SkBp1sBnD,ClBo1sBH,C;MACX,QAAQ,C;MACQ,
OAtRL,KAsRK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAxRqB,GAwRP,UAAK,UAAL,EAAK,kBAAL,SAxRO,EAwRI,OAxRJ,CAwRrB,C;;MAxRT,OA0RO,I;K;IAvRX,kC;
MAqSoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBt2sBJ,MAAO,KlBs2sBsB,wBA5RzB,KA4RyB,EAAwB,EA
AxB,CkBt2sBtB,ElBs2sBmD,SkBt2sBnD,ClBs2sBH,C;MACX,QAAQ,C;MACQ,OA9RL,KA8RK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhSqB,GAgSP,UAAK,
UAAL,EAAK,kBAAL,SAhSO,EAgSI,OAhSJ,CAgSrB,C;;MAhST,OAkSO,I;K;IA/RX,kC;MA6TqB,gB;MAHjB,gBA
AgB,gB;MAChB,WAAW,iBkBx4sBJ,MAAO,KlBw4sBsB,wBApSzB,KAoSyB,EAAwB,EAAxB,CkBx4sBtB,ElBw4sBmD,
SkBx4sBnD,ClBw4sBH,C;MACX,QAAQ,C;MACQ,OAtSL,KAsSK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxSqB,GAwSP,UAAK,UAAL,EAAK,kBAAL,SAxSO,EAwSI,O
AxSJ,CAwSrB,C;;MAxST,OA0SO,I;K;IAvSX,kC;MAqUoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB56sBJ,MAAO,
KlB46sBsB,wBA5SzB,KA4SyB,EAAwB,EAAxB,CkB56sBtB,ElB46sBmD,SkB56sBnD,ClB46sBH,C;MACX,QAAQ,C;MACQ,
OA9SL,KA8SK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAhTqB,GAgTP,UAAK,UAAL,EAAK,kBAAL,SAhTO,EAgTI,OAhTJ,CAgTrB,C;;MAhTT,OAkTO,I;K;IA/SX,kC;
MA6UoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBh8sBJ,MAAO,KlBg8sBsB,wBApTzB,KAoTyB,EAAwB,EA
AxB,CkBh8sBtB,ElBg8sBmD,SkBh8sBnD,ClBg8sBH,C;MACX,QAAQ,C;MACQ,OAtTL,KAsTK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxTqB,GAwTP,UAAK,
UAAL,EAAK,kBAAL,SAxTO,EAwTI,OAxTJ,CAwTrB,C;;MAxTT,OA0TO,I;K;IAvTX,kC;MAqVoB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBt9sBJ,MAAO,KlBs9sBsB,wBA5TzB,KA4TyB,EAAwB,EAAxB,CkBt9sBtB,ElBs9sBmD,
SkBt9sBnD,ClBs9sBH,C;MACX,QAAQ,C;MACQ,OA9TL,KA8TK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhUqB,GAgUP,UAAK,UAAL,EAAK,kBAAL,SAhUO,EAgUI,O
AhUJ,CAgUrB,C;;MAhUT,OAkUO,I;K;IA/TX,kC;MA6VoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBp+sBJ,MAAO,
KlBo+sBsB,wBApUzB,KAoUyB,EAAwB,EAAxB,CkBp+sBtB,ElBo+sBmD,SkBp+sBnD,ClBo+sBH,C;MACX,QAAQ,C;MACQ,
OAtUL,KAsUK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAxUqB,GAwUP,UAAK,UAAL,EAAK,kBAAL,SAxUO,EAwUI,OAxUJ,CAwUrB,C;;MAxUT,OA0UO,I;K;IAvUX,kC;
MAqWoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB1/sBJ,MAAO,KlB0/sBsB,wBA5UzB,KA4UyB,EAAwB,EA
AxB,CkB1/sBtB,ElB0/sBmD,SkB1/sBnD,ClB0/sBH,C;MACX,QAAQ,C;MACQ,OA9UL,KA8UK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhVqB,GAgVP,UAAK,
UAAL,EAAK,kBAAL,SAhVO,EAgVI,OAhVJ,CAgVrB,C;;MAhVT,OAkVO,I;K;IA/UX,kC;MA6WoB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBthtBJ,MAAO,KlBshtBsB,wBApVzB,KAoVyB,EAAwB,EAAxB,CkBthtBtB,ElBshtBmD,
SkBthtBnD,ClBshtBH,C;MACX,QAAQ,C;MACQ,OAtVL,KAsVK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxVqB,GAwVP,UAAK,UAAL,EAAK,kBAAL,SAxVO,EAwVI,O
AxVJ,CAwVrB,C;;MAxVT,OA0VO,I;K;IAvVX,kC;MAqXoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB5itBJ,MAAO,
KlB4itBsB,wBA5VzB,KA4VyB,EAAwB,EAAxB,CkB5itBtB,ElB4itBmD,SkB5itBnD,ClB4itBH,C;MACX,QAAQ,C;MACQ,
OA9VL,KA8VK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAhWqB,GAgWP,UAAK,UAAL,EAAK,kBAAL,SAhWO,EAgWI,OAhWJ,CAgWrB,C;;MAhWT,OAkWO,I;K;IA/VX,kC;
MA6XoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBlktBJ,MAAO,KlBkktBsB,wBApWzB,KAoWyB,EAAwB,EA
AxB,CkBlktBtB,ElBkktBmD,SkBlktBnD,ClBkktBH,C;MACX,QAAQ,C;MACQ,OAtWL,KAsWK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxWqB,GAwWP,UAAK,
UAAL,EAAK,kBAAL,SAxWO,EAwWI,OAxWJ,CAwWrB,C;;MAxWT,OA0WO,I;K;IAvWX,kC;MAqYoB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBtltBJ,MAAO,KlBsltBsB,wBA5WzB,KA4WyB,EAAwB,EAAxB,CkBtltBtB,ElBsltBmD,
SkBtltBnD,ClBsltBH,C;MACX,QAAQ,C;MACQ,OA9WL,KA8WK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhXqB,GAgXP,UAAK,UAAL,EAAK,kBAAL,SAhXO,EAgXI,O
AhXJ,CAgXrB,C;;MAhXT,OAkXO,I;K;IA/WX,kC;MA6YoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBpmtBJ,MAAO,
KlBomtBsB,wBApXzB,KAoXyB,EAAwB,EAAxB,CkBpmtBtB,ElBomtBmD,SkBpmtBnD,ClBomtBH,C;MACX,QAAQ,C;MACQ,
OAtXL,KAsXK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAxXqB,GAwXP,UAAK,UAAL,EAAK,kBAAL,SAxXO,EAwXI,OAxXJ,CAwXrB,C;;MAxXT,OA0XO,I;K;IAvXX,kC;
MAqZoB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBtntBJ,MAAO,KlBsntBsB,wBA5XzB,KA4XyB,EAAwB,EA
AxB,CkBtntBtB,ElBsntBmD,SkBtntBnD,ClBsntBH,C;MACX,QAAQ,C;MACQ,OA9XL,KA8XK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhYqB,GAgYP,UAAK,
UAAL,EAAK,kBAAL,SAhYO,EAgYI,OAhYJ,CAgYrB,C;;MAhYT,OAkYO,I;K;IA/XX,kC;MA6apB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBxotBJ,MAAO,KlBwotBsB,wBApYzB,KAoYyB,EAAwB,EAAxB,CkBxotBtB,ElBwotBmD,
SkBxotBnD,ClBwotBH,C;MACX,QAAQ,C;MACQ,OAtYL,KAsYK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxYqB,GAwYP,UAAK,UAAL,EAAK,kBAAL,SAxYO,EAwYI,O
AxYJ,CAwYrB,C;;MAxYT,OA0YO,I;K;IAvYX,kC;MAqapB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB1ptBJ,MAAO,
KlB0ptBsB,wBA5YzB,KA4YyB,EAAwB,EAAxB,CkB1ptBtB,ElB0ptBmD,SkB1ptBnD,ClB0ptBH,C;MACX,QAAQ,C;MACQ,
OA9YL,KA8YK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAhZqB,GAgZP,UAAK,UAAL,EAAK,kBAAL,SAhZO,EAgZI,OAhZJ,CAgZrB,C;;MAhZT,OAkZO,I;K;IA/YX,kC;
MA6bpB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB5qtBJ,MAAO,KlB4qtBsB,wBApZzB,KAoZyB,EAAwB,EA
AxB,CkB5qtBtB,ElB4qtBmD,SkB5qtBnD,ClB4qtBH,C;MACX,QAAQ,C;MACQ,OAtZL,KAsZK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxZqB,GAwZP,UAAK,
UAAL,EAAK,kBAAL,SAxZO,EAwZI,OAxZJ,CAwZrB,C;;MAxZT,OA0ZO,I;K;IAvZX,kC;MAqbpB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBtrtBJ,MAAO,KlBsrtBsB,wBA5ZzB,KA4ZyB,EAAwB,EAAxB,CkBtrtBtB,ElBsrtBmD,
SkBtrtBnD,ClBsrtBH,C;MACX,QAAQ,C;MACQ,OA9ZL,KA8ZK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhaqB,GAgaP,UAAK,UAAL,EAAK,kBAAL,SAhaO,EAgaI,O
AhaJ,CAgarB,C;;MAhaT,OAkaO,I;K;IA/ZX,kC;MA6cpB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBpstBJ,MAAO,
KlBostBsB,wBApazB,KAoayB,EAAwB,EAAxB,CkBpstBtB,ElBostBmD,SkBpstBnD,ClBostBH,C;MACX,QAAQ,C;MACQ,
OAtaL,KAsaK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAxaqB,GAwaP,UAAK,UAAL,EAAK,kBAAL,SAxaO,EAwaI,OAxaJ,CAwarB,C;;MAxaT,OA0aO,I;K;IAvaX,kC;
MAqcpB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBttBJ,MAAO,KlBstBsB,wBA5azB,KA4ayB,EAAwB,EAAxB,CkBttBtB,ElBstBmD,
SkBttBnD,ClBstBH,C;MACX,QAAQ,C;MACQ,OA9aL,KA8aK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhbqB,GAgbP,UAAK,UAAL,EAAK,kBAAL,SAhbO,EAgbI,O
AhbJ,CAgbrB,C;;MAhbT,OAkbO,I;K;IA/aX,kC;MA6dpB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBxutBJ,MAAO,
KlBwutBsB,wBApbzB,KAobyB,EAAwB,EAAxB,CkBxutBtB,ElBwutBmD,SkBxutBnD,ClBwutBH,C;MACX,QAAQ,C;MACQ,
OAtbL,KAsbK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAxbqB,GAwbP,UAAK,UAAL,EAAK,kBAAL,SAxbO,EAwbI,OAxbJ,CAwbrB,C;;MAxbT,OA0bO,I;K;IAvbX,kC;
MAqepB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB1vtBJ,MAAO,KlB0vtBsB,wBA5bzB,KA4byB,EAAwB,EA
AxB,CkB1vtBtB,ElB0vtBmD,SkB1vtBnD,ClB0vtBH,C;MACX,QAAQ,C;MACQ,OA9bL,KA8bK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhcqB,GAgcP,UAAK,
UAAL,EAAK,kBAAL,SAhcO,EAgcI,OAhcJ,CAgcrB,C;;MAhcT,OAkcO,I;K;IA/bX,kC;MA6dpB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBtwtBJ,MAAO,KlBswtBsB,wBApczB,KAocyB,EAAwB,EAAxB,CkBtwtBtB,ElBswtBmD,
SkBtwtBnD,ClBswtBH,C;MACX,QAAQ,C;MACQ,OAtcL,KAscK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxcqB,GAwcP,UAAK,UAAL,EAAK,kBAAL,SAxcO,EAwcI,O
AxcJ,CAwcrB,C;;MAxcT,OA0cO,I;K;IAvcX,kC;MAqfpB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBhytBJ,MAAO,
KlBgytBsB,wBA5czB,KA4cyB,EAAwB,EAAxB,CkBhytBtB,ElBgytBmD,SkBhytBnD,ClBgytBH,C;MACX,QAAQ,C;MACQ,
OA9cL,KA8cK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAhdqB,GAgdP,UAAK,UAAL,EAAK,kBAAL,SAhdO,EAgdI,OAhdJ,CAgdrB,C;;MAhdT,OAkdO,I;K;IA/cX,kC;
MA6epB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB10tBJ,MAAO,KlB00tBsB,wBApdzB,KAodyB,EAAwB,EA
AxB,CkB10tBtB,ElB00tBmD,SkB10tBnD,ClB00tBH,C;MACX,QAAQ,C;MACQ,OAtdL,KAsdK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxdqB,GAwdP,UAAK,
UAAL,EAAK,kBAAL,SAxdO,EAwdI,OAxdJ,CAwdrB,C;;MAxdT,OA0dO,I;K;IAvdX,kC;MAqfpB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBt1tBJ,MAAO,KlBs1tBsB,wBA5dzB,KA4dyB,EAAwB,EAAxB,CkBt1tBtB,ElBs1tBmD,
SkBt1tBnD,ClBs1tBH,C;MACX,QAAQ,C;MACQ,OA9dL,KA8dK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAheqB,GAgeP,UAAK,UAAL,EAAK,kBAAL,SAheO,EAgeI,O
AheJ,CAgerB,C;;MAheT,OAkeO,I;K;IA/dX,kC;MA6gpB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB52tBJ,MAAO,
KlB42tBsB,wBApezB,KAoeyB,EAAwB,EAAxB,CkB52tBtB,ElB42tBmD,SkB52tBnD,ClB42tBH,C;MACX,QAAQ,C;MACQ,
OAteL,KAseK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAxeqB,GAweP,UAAK,UAAL,EAAK,kBAAL,SAxeO,EAweI,OAxeJ,CAwerB,C;;MAxeT,OA0eO,I;K;IAveX,kC;
MAqhpB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkB94tBJ,MAAO,KlB84tBsB,wBA5ezB,KA4eyB,EAAwB,EA
AxB,CkB94tBtB,ElB84tBmD,SkB94tBnD,ClB84tBH,C;MACX,QAAQ,C;MACQ,OA9eL,KA8eK,W;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhfqB,GAgfP,UAAK,
UAAL,EAAK,kBAAL,SAhfO,EAgfI,OAhfJ,CAgfrB,C;;MAhfT,OAkfO,I;K;IA/eX,kC;MA6hpB,gB;MAHhB,gBA
AgB,gB;MAChB,WAAW,iBkBx6tBJ,MAAO,KlBw6tBsB,wBApfzB,KAofyB,EAAwB,EAAxB,CkBx6tBtB,ElBw6tBmD,
SkBx6tBnD,ClBw6tBH,C;MACX,QAAQ,C;MACQ,OAtfL,KAsfK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxfqB,GAwfP,UAAK,UAAL,EAAK,kBAAL,SAxfO,EAwfI,O
AxfJ,CAwfrB,C;;MAxfT,OA0fO,I;K;IAvfX,kC;MAqipB,gB;MAHhB,gBAAgB,gB;MAChB,WAAW,iBkBl8tBJ,MAAO,
KlBk8tBsB,wBA5fzB,KA4fyB,EAAwB,EAAxB,CkBl8tBtB,ElBk8tBmD,SkBl8tBnD,ClBk8tBH,C;MACX,QAAQ,C;MACQ,
OA9fL,KA8fK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK
,WAhgBqB,GAgfP,UAAK,UAAL,EAAK,kBAAL,SAhgBO,EAgfI,

AAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB/9rBA,iB;MlB+9rBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBl+rBJ,MAAO,KlBk+rBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBl+rBtB,ElBk+rBmD,SkBl+rBnD,ClBk+rBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBj/rBA,iB;MlBi/rBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBp/rBJ,MAAO,KlBo/rBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBp/rBtB,ElBo/rBmD,SkBp/rBnD,ClBo/rBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBr/rBA,iB;MlBq/rBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBx/rBJ,MAAO,KlBw/rBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBx/rBtB,ElBw/rBmD,SkBx/rBnD,ClBw/rBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBz/rBA,iB;MlBy/rBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5/rBJ,MAAO,KlB4/rBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5/rBtB,ElB4/rBmD,SkB5/rBnD,ClB4/rBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7/rBA,iB;MlB6/rBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhgsBJ,MAAO,KlBggsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhgsBtB,ElBggsBmD,SkBhgsBnD,ClBggsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjgsBA,iB;MlBigsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBpgsBJ,MAAO,KlBogsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBpgsBtB,ElBogsBmD,SkBpgsBnD,ClBogsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrgsBA,iB;MlBqgsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxgsBJ,MAAO,KlBwgsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxgsBtB,ElBwgsBmD,SkBxgsBnD,ClBwgsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBzgsBA,iB;MlBygsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5gsBJ,MAAO,KlB4gsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5gsBtB,ElB4gsBmD,SkB5gsBnD,ClB4gsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7gsBA,iB;MlB6gsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhhsBJ,MAAO,KlBghsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhhsBtB,ElBghsBmD,SkBhhsBnD,ClBghsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjhsBA,iB;MlBihsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBphsBJ,MAAO,KlBohsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBphsBtB,ElBohsBmD,SkBphsBnD,ClBohsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrhsBA,iB;MlBqhsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxhsBJ,MAAO,KlBwhsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxhsBtB,ElBwhsBmD,SkBxhsBnD,ClBwhsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBzhsBA,iB;MlByhsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5hsBJ,MAAO,KlB4hsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5hsBtB,ElB4hsBmD,SkB5hsBnD,ClB4hsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7hsBA,iB;MlB6hsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhisBJ,MAAO,KlBgisBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhisBtB,ElBgisBmD,SkBhisBnD,ClBgisBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjisBA,iB;MlBiisBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBpisBJ,MAAO,KlBoisBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBpisBtB,ElBoisBmD,SkBpisBnD,ClBoisBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrisBA,iB;MlBqisBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxisBJ,MAAO,KlBwisBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxisBtB,ElBwisBmD,SkBxisBnD,ClBwisBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBzisBA,iB;MlByisBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5isBJ,MAAO,KlB4isBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5isBtB,ElB4isBmD,SkB5isBnD,ClB4isBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7isBA,iB;MlB6isBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhjsBJ,MAAO,KlBgjsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhjsBtB,ElBgjsBmD,SkBhjsBnD,ClBgjsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjjsBA,iB;MlBijsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBpjsBJ,MAAO,KlBojsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBpjsBtB,ElBojsBmD,SkBpjsBnD,ClBojsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrjsBA,iB;MlBqjsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxjsBJ,MAAO,KlBwjsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxjsBtB,ElBwjsBmD,SkBxjsBnD,ClBwjsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBzjsBA,iB;MlByjsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5jsBJ,MAAO,KlB4jsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5jsBtB,ElB4jsBmD,SkB5jsBnD,ClB4jsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7jsBA,iB;MlB6jsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhksBJ,MAAO,KlBgksBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhksBtB,ElBgksBmD,SkBhksBnD,ClBgksBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjksBA,iB;MlBiksBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBpksBJ,MAAO,KlBoksBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBpksBtB,ElBoksBmD,SkBpksBnD,ClBoksBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrksBA,iB;MlBqksBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxksBJ,MAAO,KlBwksBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxksBtB,ElBwksBmD,SkBxksBnD,ClBwksBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBzksBA,iB;MlByksBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5ksBJ,MAAO,KlB4ksBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5ksBtB,ElB4ksBmD,SkB5ksBnD,ClB4ksBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7ksBA,iB;MlB6ksBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhlsBJ,MAAO,KlBglsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhlsBtB,ElBglsBmD,SkBhlsBnD,ClBglsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjlsBA,iB;MlBilsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBplsBJ,MAAO,KlBolsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBplsBtB,ElBolsBmD,SkBplsBnD,ClBolsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrlsBA,iB;MlBqlsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxlsBJ,MAAO,KlBwlsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxlsBtB,ElBwlsBmD,SkBxlsBnD,ClBwlsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBzlsBA,iB;MlBylsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5lsBJ,MAAO,KlB4lsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5lsBtB,ElB4lsBmD,SkB5lsBnD,ClB4lsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7lsBA,iB;MlB6lsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhmsBJ,MAAO,KlBgmsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhmsBtB,ElBgmsBmD,SkBhmsBnD,ClBgmsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjmsBA,iB;MlBimsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBpmsBJ,MAAO,KlBomsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBpmsBtB,ElBomsBmD,SkBpmsBnD,ClBomsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrmsBA,iB;MlBqmsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxmsBJ,MAAO,KlBwmsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxmsBtB,ElBwmsBmD,SkBxmsBnD,ClBwmsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBzmsBA,iB;MlBymsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5msBJ,MAAO,KlB4msBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5msBtB,ElB4msBmD,SkB5msBnD,ClB4msBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7msBA,iB;MlB6msBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhnsBJ,MAAO,KlBgnsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhnsBtB,ElBgnsBmD,SkBhnsBnD,ClBgnsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjnsBA,iB;MlBinsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBpnsBJ,MAAO,KlBonsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBpnsBtB,ElBonsBmD,SkBpnsBnD,ClBonsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrnsBA,iB;MlBqnsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxnsBJ,MAAO,KlBwnsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxnsBtB,ElBwnsBmD,SkBxnsBnD,ClBwnsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBznsBA,iB;MlBynsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5nsBJ,MAAO,KlB4nsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5nsBtB,ElB4nsBmD,SkB5nsBnD,ClB4nsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7nsBA,iB;MlB6nsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhosBJ,MAAO,KlBgosBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhosBtB,ElBgosBmD,SkBhosBnD,ClBgosBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjosBA,iB;MlBiosBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBposBJ,MAAO,KlBosBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBpsBtB,ElBosBmD,SkBpsBnD,ClBosBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrosBA,iB;MlBqosBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxosBJ,MAAO,KlBwosBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxosBtB,ElBwosBmD,SkBxosBnD,ClBwosBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBzosBA,iB;MlByosBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5osBJ,MAAO,KlB4osBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5osBtB,ElB4osBmD,SkB5osBnD,ClB4osBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7osBA,iB;MlB6osBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhpsBJ,MAAO,KlBgpsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhpsBtB,ElBgpsBmD,SkBhpsBnD,ClBgpsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjpsBA,iB;MlBipsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBppsBJ,MAAO,KlBopsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBppsBtB,ElBopsBmD,SkBppsBnD,ClBopsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrpsBA,iB;MlBqpsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxpsBJ,MAAO,KlBwpsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxpsBtB,ElBwpsBmD,SkBxpsBnD,ClBwpsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBzpsBA,iB;MlBypsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekB5psBJ,MAAO,KlB4psBsB,wBAAN,KAAM,EAAwB,EAAxB,CkB5psBtB,ElB4psBmD,SkB5psBnD,ClB4psBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkB7psBA,iB;MlB6psBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBhqsBJ,MAAO,KlBgqsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBhqsBtB,ElBgqsBmD,SkBhqsBnD,ClBgqsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBjqsBA,iB;MlBiqsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBpqsBJ,MAAO,KlBoqsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBpqsBtB,ElBoqsBmD,SkBpqsBnD,ClBoqsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAfX,C;+EAkBA,yB;MAAA,kF;MAAA,gE;MkBrqsBA,iB;MlBqqsBA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QAChB,WAAW,ekBxqsBJ,MAAO,KlBwqsBsB,wBAAN,KAAM,EAAwB,EAAxB,CkBxqsBtB,ElBwqsBmD,SkBxqsBnD,ClBwqsBH,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;;

,iB;MlBqqsBA,8C;QAQI,WkBvqsBO,MAAO,KlBuqsBG,gBkBvqsBH,ElBuqsBS,KAAM,OkBvqsBf,C;QlBwqsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrrsBA,iB;MlBqrsBA,8C;QAQI,WkBvrsBO,MAAO,KlBursBG,gBkBvrsBH,ElBursBS,KAAM,OkBvrsBf,C;QlBwrsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrssBA,iB;MlBqssBA,8C;QAQI,WkBvssBO,MAAO,KlBussBG,gBkBvssBH,ElBussBS,KAAM,OkBvssBf,C;QlBwssBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrtsBA,iB;MlBqtsBA,8C;QAQI,WkBvtsBO,MAAO,KlButsBG,gBkBvtsBH,ElButsBS,KAAM,OkBvtsBf,C;QlBwtsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrusBA,iB;MlBqusBA,8C;QAQI,WkBvusBO,MAAO,KlBusuBG,gBkBvusBH,ElBusuBS,KAAM,OkBvusBf,C;QlBwusBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrvsBA,iB;MlBqvsBA,8C;QAQI,WkBvvsBO,MAAO,KlBuvsBG,gBkBvvsBH,ElBuvsBS,KAAM,OkBvvsBf,C;QlBwvsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrwsBA,iB;MlBqwsBA,8C;QAQI,WkBvwsBO,MAAO,KlBuwsBG,gBkBvwsBH,ElBuwsBS,KAAM,OkBvwsBf,C;QlBwwsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrxsBA,iB;MlBqxsBA,8C;QAQI,WkBvxsBO,MAAO,KlBuxsBG,gBkBvxsBH,ElBuxsBS,KAAM,OkBvxsBf,C;QlBwxsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrysBA,iB;MlBqysBA,8C;QAQI,WkBvysBO,MAAO,KlBuysBG,gBkBvysBH,ElBuysBS,KAAM,OkBvysBf,C;QlBwysBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrzsBA,iB;MlBqzsBA,8C;QAQI,WkBvzsBO,MAAO,KlBuzsBG,gBkBvzsBH,ElBuzsBS,KAAM,OkBvzsBf,C;QlBwzsBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr0sBA,iB;MlBq0sBA,8C;QAQI,WkBv0sBO,MAAO,KlBu0sBG,gBkBv0sBH,ElBu0sBS,KAAM,OkBv0sBf,C;QlBw0sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr1sBA,iB;MlBq1sBA,8C;QAQI,WkBv1sBO,MAAO,KlBu1sBG,gBkBv1sBH,ElBu1sBS,KAAM,OkBv1sBf,C;QlBw1sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr2sBA,iB;MlBq2sBA,8C;QAQI,WkBv2sBO,MAAO,KlBu2sBG,gBkBv2sBH,ElBu2sBS,KAAM,OkBv2sBf,C;QlBw2sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr3sBA,iB;MlBq3sBA,8C;QAQI,WkBv3sBO,MAAO,KlBu3sBG,gBkBv3sBH,ElBu3sBS,KAAM,OkBv3sBf,C;QlBw3sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr4sBA,iB;MlBq4sBA,8C;QAQI,WkBv4sBO,MAAO,KlBu4sBG,gBkBv4sBH,ElBu4sBS,KAAM,OkBv4sBf,C;QlBw4sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr5sBA,iB;MlBq5sBA,8C;QAQI,WkBv5sBO,MAAO,KlBu5sBG,gBkBv5sBH,ElBu5sBS,KAAM,OkBv5sBf,C;QlBw5sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr6sBA,iB;MlBq6sBA,8C;QAQI,WkBv6sBO,MAAO,KlBu6sBG,gBkBv6sBH,ElBu6sBS,KAAM,OkBv6sBf,C;QlBw6sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr7sBA,iB;MlBq7sBA,8C;QAQI,WkBv7sBO,MAAO,KlBu7sBG,gBkBv7sBH,ElBu7sBS,KAAM,OkBv7sBf,C;QlBw7sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr8sBA,iB;MlBq8sBA,8C;QAQI,WkBv8sBO,MAAO,KlBu8sBG,gBkBv8sBH,ElBu8sBS,KAAM,OkBv8sBf,C;QlBw8sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr9sBA,iB;MlBq9sBA,8C;QAQI,WkBv9sBO,MAAO,KlBu9sBG,gBkBv9sBH,ElBu9sBS,KAAM,OkBv9sBf,C;QlBw9sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr+sBA,iB;MlBq+sBA,8C;QAQI,WkBv+sBO,MAAO,KlBu+sBG,gBkBv+sBH,ElBu+sBS,KAAM,OkBv+sBf,C;QlBw+sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr/sBA,iB;MlBq/sBA,8C;QAQI,WkBv/sBO,MAAO,KlBu/sBG,gBkBv/sBH,ElBu/sBS,KAAM,OkBv/sBf,C;QlBw/sBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrgtBA,iB;MlBqgtBA,8C;QAQI,WkBvgtBO,MAAO,KlBugtBG,gBkBvgtBH,ElBugtBS,KAAM,OkBvgtBf,C;QlBwgtBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrhtBA,iB;MlBqhtBA,8C;QAQI,WkBvhtBO,MAAO,KlBuhtBG,gBkBvhtBH,ElBuhtBS,KAAM,OkBvhtBf,C;QlBwhtBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBritBA,iB;MlBqitBA,8C;QAQI,WkBvitBO,MAAO,KlBuitBG,gBkBvitBH,ElBuitBS,KAAM,OkBvitBf,C;QlBwitBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrjtBA,iB;MlBqjtBA,8C;QAQI,WkBvjtBO,MAAO,KlBujtBG,gBkBvjtBH,ElBujtBS,KAAM,OkBvjtBf,C;QlBwjtBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrktBA,iB;MlBqktBA,8C;QAQI,WkBvktBO,MAAO,KlBuktBG,gBkBvktBH,ElBuktBS,KAAM,OkBvktBf,C;QlBwktBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrltBA,iB;MlBqltBA,8C;QAQI,WkBvltBO,MAAO,KlBultBG,gBkBvltBH,ElBultBS,KAAM,OkBvltBf,C;QlBwltBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrmtBA,iB;MlBqmtBA,8C;QAQI,WkBvmtBO,MAAO,KlBumtBG,gBkBvmtBH,ElBumtBS,KAAM,OkBvmtBf,C;QlBwmtBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrntBA,iB;MlBqntBA,8C;QAQI,WkBvntBO,MAAO,KlBuntBG,gBkBvntBH,ElBuntBS,KAAM,OkBvntBf,C;QlBwntBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrotBA,iB;MlBqotBA,8C;QAQI,WkBvotBO,MAAO,KlBuotBG,gBkBvotBH,ElBuotBS,KAAM,OkBvotBf,C;QlBwotBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrptBA,iB;MlBqptBA,8C;QAQI,WkBvptBO,MAAO,KlBuptBG,gBkBvptBH,ElBuptBS,KAAM,OkBvptBf,C;QlBwptBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrqtBA,iB;MlBqqtBA,8C;QAQI,WkBvqtBO,MAAO,KlBuqtBG,gBkBvqtBH,ElBuqtBS,KAAM,OkBvqtBf,C;QlBwqtBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrrtBA,iB;MlBqrtBA,8C;QAQI,WkBvrtBO,MAAO,KlBurtBG,gBkBvrtBH,ElBurtBS,KAAM,OkBvrtBf,C;QlBwrtBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrstBA,iB;MlBqstBA,8C;QAQI,WkBvstBO,MAAO,KlBustBG,gBkBvstBH,ElBustBS,KAAM,OkBvstBf,C;QlBwstBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrttBA,iB;MlBqttBA,8C;QAQI,WkBvttBO,MAAO,KlButtBG,gBkBvttBH,ElButtBS,KAAM,OkBvttBf,C;QlBwttBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrutBA,iB;MlBqutBA,8C;QAQI,WkBvutBO,MAAO,KlBuutBG,gBkBvutBH,ElBuutBS,KAAM,OkBvutBf,C;QlBwutBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrvtBA,iB;MlBqvtBA,8C;QAQI,WkBvvtBO,MAAO,KlBuvtBG,gBkBvvtBH,ElBuvtBS,KAAM,OkBvvtBf,C;QlBwvtBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrwtBA,iB;MlBqwtBA,8C;QAQI,WkBvwtBO,MAAO,KlBuwtBG,gBkBvwtBH,ElBuwtBS,KAAM,OkBvwtBf,C;QlBwwtBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrxtBA,iB;MlBqxtBA,8C;QAQI,WkBvxtBO,MAAO,KlBuxtBG,gBkBvxtBH,ElBuxtBS,KAAM,OkBvxtBf,C;QlBwxtBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrytBA,iB;MlBqytBA,8C;QAQI,WkBvytBO,MAAO,KlBuytBG,gBkBvytBH,ElBuytBS,KAAM,OkBvytBf,C;QlBwytBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrztBA,iB;MlBqztBA,8C;QAQI,WkBvztBO,MAAO,KlBuztBG,gBkBvztBH,ElBuztBS,KAAM,OkBvztBf,C;QlBwztBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr0tBA,iB;MlBq0tBA,8C;QAQI,WkBv0tBO,MAAO,KlBu0tBG,gBkBv0tBH,ElBu0tBS,KAAM,OkBv0tBf,C;QlBw0tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr1tBA,iB;MlBq1tBA,8C;QAQI,WkBv1tBO,MAAO,KlBu1tBG,gBkBv1tBH,ElBu1tBS,KAAM,OkBv1tBf,C;QlBw1tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr2tBA,iB;MlBq2tBA,8C;QAQI,WkBv2tBO,MAAO,KlBu2tBG,gBkBv2tBH,ElBu2tBS,KAAM,OkBv2tBf,C;QlBw2tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr3tBA,iB;MlBq3tBA,8C;QAQI,WkBv3tBO,MAAO,KlBu3tBG,gBkBv3tBH,ElBu3tBS,KAAM,OkBv3tBf,C;QlBw3tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr4tBA,iB;MlBq4tBA,8C;QAQI,WkBv4tBO,MAAO,KlBu4tBG,gBkBv4tBH,ElBu4tBS,KAAM,OkBv4tBf,C;QlBw4tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr5tBA,iB;MlBq5tBA,8C;QAQI,WkBv5tBO,MAAO,KlBu5tBG,gBkBv5tBH,ElBu5tBS,KAAM,OkBv5tBf,C;QlBw5tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr6tBA,iB;MlBq6tBA,8C;QAQI,WkBv6tBO,MAAO,KlBu6tBG,gBkBv6tBH,ElBu6tBS,KAAM,OkBv6tBf,C;QlBw6tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr7tBA,iB;MlBq7tBA,8C;QAQI,WkBv7tBO,MAAO,KlBu7tBG,gBkBv7tBH,ElBu7tBS,KAAM,OkBv7tBf,C;QlBw7tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr8tBA,iB;MlBq8tBA,8C;QAQI,WkBv8tBO,MAAO,KlBu8tBG,gBkBv8tBH,ElBu8tBS,KAAM,OkBv8tBf,C;QlBw8tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr9tBA,iB;MlBq9tBA,8C;QAQI,WkBv9tBO,MAAO,KlBu9tBG,gBkBv9tBH,ElBu9tBS,KAAM,OkBv9tBf,C;QlBw9tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBr+tBA,iB;MlBq+tBA,8C;QAQI,WkBv+tBO,MAAO,KlBu+tBG,gBkBv+tBH,ElBu+tBS,KAAM,OkBv+tBf,C;QlBw+tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBr/tBA,iB;MlBq/tBA,8C;QAQI,WkBv/tBO,MAAO,KlBu/tBG,gBkBv/tBH,ElBu/tBS,KAAM,OkBv/tBf,C;QlBw/tBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrguBA,iB;MlBqguBA,8C;QAQI,WkBvguBO,MAAO,KlBuguBG,gBkBvguBH,ElBuguBS,KAAM,OkBvguBf,C;QlBwguBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrhuBA,iB;MlBqhuBA,8C;QAQI,WkBvhuBO,MAAO,KlBuhuBG,gBkBvhuBH,ElBuhuBS,KAAM,OkBvhuBf,C;QlBwhuBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBritBA,iB;MlBqitBA,8C;QAQI,WkBvitBO,MAAO,KlBuitBG,gBkBvitBH,ElBuitBS,KAAM,OkBvitBf,C;QlBwitBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrjuBA,iB;MlBqjuBA,8C;QAQI,WkBvjuBO,MAAO,KlBujuBG,gBkBvjuBH,ElBujuBS,KAAM,OkBvjuBf,C;QlBwjuBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrktBA,iB;MlBqktBA,8C;QAQI,WkBvktBO,MAAO,KlBuktBG,gBkBvktBH,ElBuktBS,KAAM,OkBvktBf,C;QlBwktBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrltBA,iB;MlBqltBA,8C;QAQI,WkBvltBO,MAAO,KlBultBG,gBkBvltBH,ElBultBS,KAAM,OkBvltBf,C;QlBwltBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrmtBA,iB;MlBqmtBA,8C;QAQI,WkBvmtBO,MAAO,KlBumtBG,gBkBvmtBH,ElBumtBS,KAAM,OkBvmtBf,C;QlBwmtBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrntBA,iB;MlBqntBA,8C;QAQI,WkBvntBO,MAAO,KlBuntBG,gBkBvntBH,ElBuntBS,KAAM,OkBvntBf,C;QlBwntBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MkBrotBA,iB;MlBqotBA,8C;QAQI,WkBvotBO,MAAO,KlBuotBG,gBkBvotBH,ElBuotBS,KAAM,OkBvotBf,C;QlBwotBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAnB,CAAJ,C;;QAET,OAAO,I;O;KAbX,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBrptBA,iB;MlBqptBA,8C;QAQI,WkBvptBO,MAAO,KlBuptBG,gBkBvptBH,ElBuptBS,KAAM,OkBvptBf,C;QlBwptBd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,

AA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAyC,I;MAGxN,Q;MAFhB,MAAO,gBAAO,MAAP,C;MACP,YAAY,C;MACZ,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,IAAI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,CAAR,IAAa,SAAS,KAA1B,C;UACI,IAAI,iBAAJ,C;YACI,MAAO,gBAAO,UAAU,OAAV,CAAP,C;;YAEP,MAAO,gBAAO,OAAQ,WAAf,C;;UACR,K;;MAEX,IAAI,SAAS,CAAT,IAAc,SAAS,KAA3B,C;UACI,IAAI,iBAAJ,C;YACI,MAAO,gBAAO,SAAP,C;QACX

C;O;K;IAL3B,iC;MAII,IAlleO,qBAAQ,CAklef,C;QAAe,OAAO,W;MACtB,kCAAgB,8BAAhB,C;K;IAQgB,8C;MAAA,mB;QAAE,OAAK,wBAAL,eAAK,C;O;K;IAL3B,iC;MAII,IAlleO,qBAAQ,CAklef,C;QAAe,OAAO,W;MACtB,kCAAgB,8BAAhB,C;K;IAQgB,8C;MAAA,mB;QAAE,OAAK,yBAAL,eAAK,C;O;K;IAL3B,iC;MAII,IAlleO,qBAAQ,CAklef,C;QAAe,OAAO,W;MACtB,kCAAgB,8BAAhB,C;K;IAQgB,8C;MAAA,mB;QAAE,OAAK,0BAAL,eAAK,C;O;K;IAL3B,iC;MAII,IAlleO,qBAAQ,CAklef,C;QAAe,OAAO,W;MACtB,kCAAgB,8BAAhB,C;K;IAQgB,8C;MAAA,mB;QAAE,OAAK,2BAAL,eAAK,C;O;K;IAL3B,iC;MAII,IAlleO,qBAAQ,CAklef,C;QAAe,OAAO,W;MACtB,kCAAgB,8BAAhB,C;K;IAQgB,8C;MAAA,mB;QAAE,OAAK,4BAAL,eAAK,C;O;K;IAL3B,iC;MAII,IAlleO,qBAAQ,CAklef,C;QAAe,OAAO,W;MACtB,kCAAgB,8BAAhB,C;K;IAQgB,8C;MAAA,mB;QAAE,OAAK,yBAAL,eAAK,C;O;K;IAL3B,iC;MAII,IAlleO,qBAAQ,CAklef,C;QAAe,OAAO,W;MACtB,kCAAgB,8BAAhB,C;K;IAUgB,4C;MAAA,mB;QAAE,OAAK,qBAAL,eAAK,C;O;K;IAP3B,+B;MAMI,IA5peO,qBAAQ,CA4pef,C;QAAe,OAAO,e;MACtB,kCAAgB,4BAAhB,C;K;IAUgB,8C;MAAA,mB;QAAE,OAAK,yBAAL,eAAK,C;O;K;IAP3B,iC;MAMI,IA9peO,qBAAQ,CA8pef,C;QAAe,OAAO,e;MACtB,kCAAgB,8BAAhB,C;K;IAUgB,8C;MAAA,mB;QAAE,OAAK,0BAAL,eAAK,C;O;K;IAP3B,iC;MAMI,IAhqeO,qBAAQ,CAgqef,C;QAAe,OAAO,e;MACtB,kCAAgB,8BAAhB,C;K;IAUgB,8C;MAAA,mB;QAAE,OAAK,wBAAL,eAAK,C;O;K;IAP3B,iC;MAMI,IAlqeO,qBAAQ,CAkqef,C;QAAe,OAAO,e;MACtB,kCAAgB,8BAAhB,C;K;IAUgB,8C;MAAA,mB;QAAE,OAAK,yBAAL,eAAK,C;O;K;IAP3B,iC;MAMI,IApqeO,qBAAQ,CAoqef,C;QAAe,OAAO,e;MACtB,kCAAgB,8BAAhB,C;K;IAUgB,8C;MAAA,mB;QAAE,OAAK,0BAAL,eAAK,C;O;K;IAP3B,iC;MAMI,IAtqeO,qBAAQ,CAsqef,C;QAAe,OAAO,e;MACtB,kCAAgB,8BAAhB,C;K;IAUgB,8C;MAAA,mB;QAAE,OAAK,2BAAL,eAAK,C;O;K;IAP3B,iC;MAMI,IAxqeO,qBAAQ,CAwqef,C;QAAe,OAAO,e;MACtB,kCAAgB,8BAAhB,C;K;IAUgB,8C;MAAA,mB;QAAE,OAAK,4BAAL,eAAK,C;O;K;IAP3B,iC;MAMI,IA1qeO,qBAAQ,CA0qef,C;QAAe,OAAO,e;MACtB,kCAAgB,8BAAhB,C;K;IAUgB,8C;MAAA,mB;QAAE,OAAK,yBAAL,eAAK,C;O;K;IAP3B,iC;MAMI,IA5qeO,qBAAQ,CA4qef,C;QAAe,OAAO,e;MACtB,kCAAgB,8BAAhB,C;K;IAGJ,4B;MAOO,Q;MAFhB,UAAkB,G;MAClB,YAAiB,C;MACjB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OAAO,O;QACP,qB;;MAEJ,OAAW,UAAS,CAAb,gB,GAAgB,wCAAO,IAAvB,vB,GAAuB,gBAAvB,gCAAuB,KAAvB,K;IAGjD,8

Cf,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,YAAO,OAAP,I;;MAEJ,OAAO,G;K;IAGX,0B;MAMo
B,Q;MADhB,Y;MACA,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,cAAO,OAAP,C;;MAEJ,OAAO,G
;K;IAGX,0B;MAMoB,Q;MADhB,UAAiB,G;MACjB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OA
AO,O;;MAEX,OAAO,G;K;IAGX,0B;MAMoB,Q;MADhB,UAAkB,G;MAClB,wBAAgB,SAAhB,gB;QAAgB,cA
AA,SAAhB,M;QACI,OAAO,O;;MAEX,OAAO,G;K;IAGX,0B;MAKoB,Q;MADhB,UAAe,C;MACf,wBAAgB,SA
AhB,gB;QAAgB,cAAA,SAAhB,M;QACI,YAAO,O;;MAEX,OAAO,G;K;IAGX,0B;MAKoB,Q;MADhB,UAAe,C;
MACf,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,YAAO,O;;MAEX,OAAO,G;K;IAGX,0B;MAKoB,
Q;MADhB,UAAe,C;MACf,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,YAAO,OAAP,I;;MAEJ,OAA
O,G;K;IAGX,0B;MAKoB,Q;MADhB,Y;MACA,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,cAAO,O
AAP,C;;MAEJ,OAAO,G;K;IAGX,0B;MAKoB,Q;MADhB,UAAiB,G;MACjB,wBAAgB,SAAhB,gB;QAAgB,cAA
A,SAAhB,M;QACI,OAAO,O;;MAEX,OAAO,G;K;IAGX,2B;MAKoB,Q;MADhB,UAAkB,G;MAClB,wBAAgB,S
AAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OAAO,O;;MAEX,OAAO,G;K;Ia5uuBX,oD;MAQuF,wC;K;IARvF,8C
ASI,Y;MAAuC,8B;K;IAT3C,gF;4FOOA,qB;MAOI,OAAO,sBAAI,CAAJ,C;K;4FAGX,qB;MAOI,OAAO,sBAAI,
CAAJ,C;K;4FAGX,qB;MAOI,OAAO,sBAAI,CAAJ,C;K;4FAGX,qB;MAOI,OAAO,sBAAI,CAAJ,C;K;4FAGX,q
B;MAOI,OAAO,sBAAI,CAAJ,C;K;IAGX,wC;MAII,IAAI,oCAAJ,C;QACI,OAAO,yBAAS,OAAT,C;MACX,OA
AO,qBAAQ,OAAR,KAAoB,C;K;IAWG,yC;MAAA,qB;QAAE,MAAM,8BAA0B,iDAA8C,aAA9C,MAA1B,C;O;
K;IAR1C,qC;MAMI,IAAI,8BAAJ,C;QACI,OAAO,sBAAI,KAAJ,C;MACX,OAAO,6BAAgB,KAAhB,EAAuB,uB
AAvB,C;K;0FAGX,4B;MAOI,OAAO,sBAAI,KAAJ,C;K;IAGX,2D;MAcqB,Q;MARjB,IAAI,8BAAJ,C;QACI,OA
AsB,KA4Lf,IAAS,CAAT,IA5Le,KA4LD,IAAS,iBA5LvB,SA4LuB,CAA3B,GA5LI,SA4LkC,aA5LnB,KA4LmB,C
AAtC,GA5L0B,YA4L4B,CA5LnC,KA4LmC,C;;MA3L7D,IAAI,QAAQ,CAAZ,C;QACI,OAAO,aAAa,KAAb,C;M
ACX,eAAe,oB;MACf,YAAY,C;MACZ,OAAO,QAAS,UAAhB,C;QACI,cAAc,QAAS,O;QACvB,IAAI,WAAS,Y
AAT,EAAS,oBAAT,OAAJ,C;UACI,OAAO,O;;MAEf,OAAO,aAAa,KAAb,C;K;sGAGX,yB;MAAA,8D;MAAA,i
D;QAOI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,sBAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;K
APjE,C;IAUA,6C;MAcqB,Q;MARjB,IAAI,8BAAJ,C;QACI,OAAY,YAAL,SAAK,EAAU,KAAV,C;MAChB,IAA
I,QAAQ,CAAZ,C;QACI,OAAO,I;MACX,eAAe,oB;MACf,YAAY,C;MACZ,OAAO,QAAS,UAAhB,C;QACI,cAA
c,QAAS,O;QACvB,IAAI,WAAS,YAAT,EAAS,oBAAT,OAAJ,C;UACI,OAAO,O;;MAEf,OAAO,I;K;sGAGX,yB;
MAAA,sD;MAAA,mC;QAOI,OAAY,UAAL,SAAK,EAAU,KAAV,C;O;KAPhB,C;gFAUA,gC;MAOW,Q;MAAP,HS,Q;QAAA,2B;QAAhB,oB,cAAgB,yB;UAAA,M,IAAgB,WAAhB,C;UAAA,yB;UAAA,M,IAAgB,WAAhB,C;UAAA,yB;UAAA,M,IAAgB,WAAhB,C;UAAA

C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,mBAAmB,KAAnB,C;QACA,IAAI,gBAAW,IAAX,CAAJ,C;UACI,OAAO,K;QACX,qB;;MAEJ,OAAO,E;K;IAGX,uC;MAKI,OAAO,wBAAQ,OAAR,C;K;gGAGX,yB;MAAA,wE;MAAA,uC;QAKiB,Q;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,mBAAmB,KAAnB,C;UACA,IAAI,UAAU,IAAV,CAAJ,C;YACI,OAAO,K;UACX,qB;;QAEJ,OAAO,E;O;KAXX,C;gGAcA,gC;MAKiB,Q;MADb,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAI,UAAU,IAAV,CAAJ,C;UACI,OAAO,K;QACX,qB;;MAEJ,OAAO,E;K;8FAGX,yB;MAAA,wE;MAAA,uC;QAMiB,Q;QAFb,gBAAgB,E;QAChB,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,mBAAmB,KAAnB,C;UACA,IAAI,UAAU,IAAV,CAAJ,C;UACI,YAAY,K;UAChB,qB;;QAEJ,OAAO,S;O;KAZX,C;8FAeA,eC;MAII,eAAe,SAAK,sBAAa,cAAb,C;MACpB,OAAO,QAAS,cAAhB,C;QACI,OAAO,QAAS,WAAhB,C;UACI,OAAO,QAAS,WAAhB,C;YACI,OAAO,QAAS,WAAhB,C;cACI,OAAO,QAAS,WAAhB,C;gBACI,OAAO,QAAS,WAAhB,C;kBACI,OAAO,QAAS,WAAhB,C;oBACI,OAAO,QAAS,WAAhB,C;sBACI,OAAO,QAAS,WAAhB,C;wBACI,OAAO,QAAS,WAAhB,C;0BACI,OAAO,QAAS,WAAhB,C;4BACI,OAAO,QAAS,WAAhB,C;8BACI,OAAO,QAAS,WAAhB,C;gCACI,OAAO,QAAS,WAAhB,C;kCACI,OAAO,QAAS,WAAhB,C;oCACI,OAAO,QAAS,WAAhB,C;sCACI,OAAO,QAAS,WAAhB,C;wCACI,OAAO,QAAS,WAAhB,C;0CACI,OAAO,QAAS,WAAhB,C;4CACI,OAAO,QAAS,WAAhB,C;8CACI,OAAO,QAAS,WAAhB,C;gDACI,OAAO,QAAS,WAAhB,C;kDACI,OAAO,QAAS,WAAhB,C;oDACI,OAAO,QAAS,WAAhB,C;sDACI,OAAO,QAAS,WAAhB,C;wDACI,OAAO,QAAS,WAAhB,C;0DACI,OAAO,QAAS,WAAhB,C;4DACI,OAAO,QAAS,WAAhB,C;8DACI,OAAO,QAAS,WAAhB,C;gEACI,OAAO,QAAS,WAAhB,C;kEACI,OAAO,QAAS,WAAhB,C;oEACI,OAAO,QAAS,WAAhB,C;sEACI,OAAO,QAAS,WAAhB,C;wEACI,OAAO,QAAS,WAAhB,C;0EACI,OAAO,QAAS,WAAhB,C;4EACI,OAAO,QAAS,WAAhB,C;8EACI,OAAO,QAAS,WAAhB,C;gFACI,OAAO,QAAS,WAAhB,C;kFACI,OAAO,QAAS,WAAhB,C;oFACI,OAAO,QAAS,WAAhB,C;sFACI,OAAO,QAAS,WAAhB,C;wFACI,OAAO,QAAS,WAAhB,C;0FACI,OAAO,QAAS,WAAhB,C;4FACI,OAAO,QAAS,WAAhB,C;8FACI,OAAO,QAAS,WAAhB,C;gGACI,OAAO,QAAS,WAAhB,C;kGACI,OAAO,QAAS,WAAhB,C;oGACI,OAAO,QAAS,WAAhB,C;sGACI,OAAO,QAAS,WAAhB,C;wGACI,OAAO,QAAS,WAAhB,C;0GACI,OAAO,QAAS,WAAhB,C;4GACI,OAAO,QAAS,WAAhB,C;8GACI,OAAO,QAAS,WAAhB,C;gHACI,OAAO,QAAS,WAAhB,C;kHACI,OAAO,QAAS,WAAhB,C;oHACI,OAAO,QAAS,WAAhB,C;sHACI,OAAO,QAAS,WAAhB,C;wHACI,OAAO,QAAS,WAAhB,C;0HACI,OAAO,QAAS,WAAhB,C;4HACI,OAAO,QAAS,WAAhB,C;8HACI,OAAO,QAAS,WAAhB,C;gIACI,OAAO,QAAS,WAAhB,C;kIACI,OAAO,QAAS,WAAhB,C;oIACI,OAAO,QAAS,WAAhB,C;sIACI,OAAO,QAAS,WAAhB,C;wIACI,OAAO,QAAS,WAAhB,C;0IACI,OAAO,QAAS,WAAhB,C;4IACI,OAAO,QAAS,WAAhB,C;8IACI,OAAO,QAAS,WAAhB,C;gJACI,OAAO,QAAS,WAAhB,C;kJACI,OAAO,QAAS,WAAhB,C;oJACI,OAAO,QAAS,WAAhB,C;sJACI,OAAO,QAAS,WAAhB,C;wJACI,OAAO,QAAS,WAAhB,C;0JACI,OAAO,QAAS,WAAhB,C;4JACI,OAAO,QAAS,WAAhB,C;8JACI,OAAO,QAAS,WAAhB,C;gKACI,OAAO,QAAS,WAAhB,C;kKACI,OAAO,QAAS,WAAhB,C;oKACI,OAAO,QAAS,WAAhB,C;sKACI,OAAO,QAAS,WAAhB,C;wKACI,OAAO,QAAS,WAAhB,C;0KACI,OAAO,QAAS,WAAhB,C;4KACI,OAAO,QAAS,WAAhB,C;8KACI,OAAO,QAAS,WAAhB,C;gLACI,OAAO,QAAS,WAAhB,C;kLACI,OAAO,QAAS,WAAhB,C;oLACI,OAAO,QAAS,WAAhB,C;sLACI,OAAO,QAAS,WAAhB,C;wLACI,OAAO,QAAS,WAAhB,C;0LACI,OAAO,QAAS,WAAhB,C;4LACI,OAAO,QAAS,WAAhB,C;8LACI,OAAO,QAAS,WAAhB,C;gMACI,OAAO,QAAS,WAAhB,C;kMACI,OAAO,QAAS,WAAhB,C;oMACI,OAAO,QAAS,WAAhB,C;sMACI,OAAO,QAAS,WAAhB,C;wMACI,OAAO,QAAS,WAAhB,C;0MACI,OAAO,QAAS,WAAhB,C;4MACI,OAAO,QAAS,WAAhB,C;8MACI,OAAO,QAAS,WAAhB,C;gNACI,OAAO,QAAS,WAAhB,C;kNACI,OAAO,QAAS,WAAhB,C;oNACI,OAAO,QAAS,WAAhB,C;sNACI,OAAO,QAAS,WAAhB,C;wNACI,OAAO,QAAS,WAAhB,C;0NACI,OAAO,QAAS,WAAhB,C;4NACI,OAAO,QAAS,WAAhB,C;8NACI,OAAO,QAAS,WAAhB,C;gOACI,OAAO,QAAS,WAAhB,C;kOACI,OAAO,QAAS,WAAhB,C;oOACI,OAAO,QAAS,WAAhB,C;sOACI,OAAO,QAAS,WAAhB,C;wOACI,OAAO,QAAS,WAAhB,C;0OACI,OAAO,QAAS,WAAhB,C;4OACI,OAAO,QAAS,WAAhB,C;8OACI,OAAO,QAAS,WAAhB,C;gPACI,OAAO,QAAS,WAAhB,C;kPACI,OAAO,QAAS,WAAhB,C;oPACI,OAAO,QAAS,WAAhB,C;sPACI,OAAO,QAAS,WAAhB,C;wPACI,OAAO,QAAS,WAAhB,C;0PACI,OAAO,QAAS,WAAhB,C;4PACI,OAAO,QAAS,WAAhB,C;8PACI,OAAO,QAAS,WAAhB,C;gQACI,OAAO,QAAS,WAAhB,C;kQACI,OAAO,QAAS,WAAhB,C;oQACI,OAAO,QAAS,WAAhB,C;sQACI,OAAO,QAAS,WAAhB,C;wQACI,OAAO,QAAS,WAAhB,C;0QACI,OAAO,QAAS,WAAhB,C;4QACI,OAAO,QAAS,WAAhB,C;8QACI,OAAO,QAAS,WAAhB,C;gRACI,OAAO,QAAS,WAAhB,C;kRACI,OAAO,QAAS,WAAhB,C;oRACI,OAAO,QAAS,WAAhB,C;sRACI,OAAO,QAAS,WAAhB,C;wRACI,OAAO,QAAS,WAAhB,C;0RACI,OAAO,QAAS,WAAhB,C;4RACI,OAAO,QAAS,WAAhB,C;8RACI,OAAO,QAAS,WAAhB,C;gSACI,OAAO,QAAS,WAAhB,C;kSACI,OAAO,QAAS,WAAhB,C;oSACI,OAAO,QAAS,WAAhB,C;sSACI,OAAO,QAAS,WAAhB,C;wSACI,OAAO,QAAS,WAAhB,C;0SACI,OAAO,QAAS,WAAhB,C;4SACI,OAAO,QAAS,WAAhB,C;8SACI,OAAO,QAAS,WAAhB,C;gTACI,OAAO,QAAS,WAAhB,C;kTACI,OAAO,QAAS,WAAhB,C;oTACI,OAAO,QAAS,WAAhB,C;sTACI,OAAO,QAAS,WAAhB,C;wTACI,OAAO,QAAS,WAAhB,C;0TACI,OAAO,QAAS,WAAhB,C;4TACI,OAAO,QAAS,WAAhB,C;8TACI,OAAO,QAAS,WAAhB,C;gUACI,OAAO,QAAS,WAAhB,C;kUACI,OAAO,QAAS,WAAhB,C;oUACI,OAAO,QAAS,WAAhB,C;sUACI,OAAO,QAAS,WAAhB,C;wUACI,OAAO,QAAS,WAAhB,C;0UACI,OAAO,QAAS,WAAhB,C;4UACI,OAAO,QAAS,WAAhB,C;8UACI,OAAO,QAAS,WAAhB,C;gVACI,OAAO,QAAS,WAAhB,C;kVACI,OAAO,QAAS,WAAhB,C;oVACI,OAAO,QAAS,WAAhB,C;sVACI,OAAO,QAAS,WAAhB,C;wVACI,OAAO,QAAS,WAAhB,C;0VACI,OAAO,QAAS,WAAhB,C;4VACI,OAAO,QAAS,WAAhB,C;8VACI,OAAO,QAAS,WAAhB,C;gWACI,OAAO,QAAS,WAAhB,C;kWACI,OAAO,QAAS,WAAhB,C;oWACI,OAAO,QAAS,WAAhB,C;sWACI,OAAO,QAAS,WAAhB,C;wWACI,OAAO,QAAS,WAAhB,C;0WACI,OAAO,QAAS,WAAhB,C;4WACI,OAAO,QAAS,WAAhB,C;8WACI,OAAO,QAAS,WAAhB,C;gXACI,OAAO,QAAS,WAAhB,C;kXACI,OAAO,QAAS,WAAhB,C;oXACI,OAAO,QAAS,WAAhB,C;sXACI,OAAO,QAAS,WAAhB,C;wXACI,OAAO,QAAS,WAAhB,C;0XACI,OAAO,QAAS,WAAhB,C;4XACI,OAAO,QAAS,WAAhB,C;8XACI,OAAO,QAAS,WAAhB,C;gYACI,OAAO,QAAS,WAAhB,C;kYACI,OAAO,QAAS,WAAhB,C;oYACI,OAAO,QAAS,WAAhB,C;sYACI,OAAO,QAAS,WAAhB,C;wYACI,OAAO,QAAS,WAAhB,C;0YACI,OAAO,QAAS,WAAhB,C;4YACI,OAAO,QAAS,WAAhB,C;8YACI,OAAO,QAAS,WAAhB,C;gZACI,OAAO,QAAS,WAAhB,C;kZACI,OAAO,QAAS,WAAhB,C;oZACI,OAAO,QAAS,WAAhB,C;sZACI,OAAO,QAAS,WAAhB,C;wZACI,OAAO,QAAS,WAAhB,C;0ZACI,OAAO,QAAS,WAAhB,C;4ZACI,OAAO,QAAS,WAAhB,C;8ZACI,OAAO,QAAS,WAAhB,C;gaACI,OAAO,QAAS,WAAhB,C;kaACI,OAAO,QAAS,WAAhB,C;oaACI,OAAO,QAAS,WAAhB,C;saACI,OAAO,QAAS,WAAhB,C;waACI,OAAO,QAAS,WAAhB,C;0aACI,OAAO,QAAS,WAAhB,C;4aACI,OAAO,QAAS,WAAhB,C;8aACI,OAAO,QAAS,WAAhB,C;gbACI,OAAO,QAAS,WAAhB,C;kbACI,OAAO,QAAS,WAAhB,C;obACI,OAAO,QAAS,WAAhB,C;sbACI,OAAO,QAAS,WAAhB,C;wbACI,OAAO,QAAS,WAAhB,C;0bACI,OAAO,QAAS,WAAhB,C;4bACI,OAAO,QAAS,WAAhB,C;8bACI,OAAO,QAAS,WAAhB,C;gcACI,OAAO,QAAS,WAAhB,C;kcACI,OAAO,QAAS,WAAhB,C;ocACI,OAAO,QAAS,WAAhB,C;scACI,OAAO,QAAS,WAAhB,C;wcACI,OAAO,QAAS,WAAhB,C;0cACI,OAAO,QAAS,WAAhB,C;4cACI,OAAO,QAAS,WAAhB,C;8cACI,OAAO,QAAS,WAAhB,C;gdACI,OAAO,QAAS,WAAhB,C;kdACI,OAAO,QAAS,WAAhB,C;odACI,OAAO,QAAS,WAAhB,C;sdACI,OAAO,QAAS,WAAhB,C;wdACI,OAAO,QAAS,WAAhB,C;0dACI,OAAO,QAAS,WAAhB,C;4dACI,OAAO,QAAS,WAAhB,C;8dACI,OAAO,QAAS,WAAhB,C;gecACI,OAAO,QAAS,WAAhB,C;kecACI,OAAO,QAAS,WAAhB,C;oecACI,OAAO,QAAS,WAAhB,C;secACI,OAAO,QAAS,WAAhB,C;wecACI,OAAO,QAAS,WAAhB,C;0ecACI,OAAO,QAAS,WAAhB,C;4ecACI,OAAO,QAAS,WAAhB,C;8ecACI,OAAO,QAAS,WAAhB,C;gfACI,OAAO,QAAS,WAAhB,C;kfACI,OAAO,QAAS,WAAhB,C;ofACI,OAAO,QAAS,WAAhB,C;sfACI,OAAO,QAAS,WAAhB,C;wfACI,OAAO,QAAS,WAAhB,C;0fACI,OAAO,QAAS,WAAhB,C;4fACI,OAAO,QAAS,WAAhB,C;8fACI,OAAO,QAAS,WAAhB,C;ggACI,OAAO,QAAS,WAAhB,C;kgACI,OAAO,QAAS,WAAhB,C;ogACI,OAAO,QAAS,WAAhB,C;sgACI,OAAO,QAAS,WAAhB,C;wgACI,OAAO,QAAS,WAAhB,C;0gACI,OAAO,QAAS,WAAhB,C;4gACI,OAAO,QAAS,WAAhB,C;8gACI,OAAO,QAAS,WAAhB,C;ghACI,OAAO,QAAS,WAAhB,C;khACI,OAAO,QAAS,WAAhB,C;ohACI,OAAO,QAAS,WAAhB,C;shACI,OAAO,QAAS,WAAhB,C;whACI,OAAO,QAAS,WAAhB,C;0hACI,OAAO,QAAS,WAAhB,C;4hACI,OAAO,QAAS,WAAhB,C;8hACI,OAAO,QAAS,WAAhB,C;giACI,OAAO,QAAS,WAAhB,C;kiACI,OAAO,QAAS,WAAhB,C;oiACI,OAAO,QAAS,WAAhB,C;siACI,OAAO,QAAS,WAAhB,C;wiACI,OAAO,QAAS,WAAhB,C;0iACI,OAAO,QAAS,WAAhB,C;4iACI,OAAO,QAAS,WAAhB,C;8iACI,OAAO,QAAS,WAAhB,C;gjACI,OAAO,QAAS,WAAhB,C;kjACI,OAAO,QAAS,WAAhB,C;ojACI,OAAO,QAAS,WAAhB,C;sjACI,OAAO,QAAS,WAAhB,C;wjACI,OAAO,QAAS,WAAhB,C;0jACI,OAAO,QAAS,WAAhB,C;4jACI,OAAO,QAAS,WAAhB,C;8jACI,OAAO,QAAS,WAAhB,C;gkACI,OAAO,QAAS,WAAhB,C;kkACI,OAAO,QAAS,WAAhB,C;okACI,OAAO,QAAS,WAAhB,C;skACI,OAAO,QAAS,WAAhB,C;wkACI,OAAO,QAAS,WAAhB,C;0kACI,OAAO,QAAS,WAAhB,C;4kACI,OAAO,QAAS,WAAhB,C;8kACI,OAAO,QAAS,WAAhB,C;glACI,OAAO,QAAS,WAAhB,C;klACI,OAAO,QAAS,WAAhB,C;olACI,OAAO,QAAS,WAAhB,C;slACI,OAAO,QAAS,WAAhB,C;wlACI,OAAO,QAAS,WAAhB,C;0lACI,OAAO,QAAS,WAAhB,C;4lACI,OAAO,QAAS,WAAhB,C;8lACI,OAAO,QAAS,WAAhB,C;gmACI,OAAO,QAAS,WAAhB,C;kmACI,OAAO,QAAS,WAAhB,C;omACI,OAAO,QAAS,WAAhB,C;smACI,OAAO,QAAS,WAAhB,C;wmACI,OAAO,QAAS,WAAhB,C;0mACI,OAAO,QAAS,WAAhB,C;4mACI,OAAO,QAAS,WAAhB,C;8mACI,OAAO,QAAS,WAAhB,C;gnACI,OAAO,QAAS,WAAhB,C;knACI,OAAO,QAAS,WAAhB,C;onACI,OAAO,QAAS,WAAhB,C;snACI,OAAO,QAAS,WAAhB,C;wnACI,OAAO,QAAS,WAAhB,C;0nACI,OAAO,QAAS,WAAhB,C;4nACI,OAAO,QAAS,WAAhB,C;8nACI,OAAO,QAAS,WAAhB,C;goACI,OAAO,QAAS,WAAhB,C;koACI,OAAO,QAAS,WAAhB,C;ooACI,OAAO,QAAS,WAAhB,C;soACI,OAAO,QAAS,WAAhB,C;woACI,OAAO,QAAS,WAAhB,C;0oACI,OAAO,QAAS,WAAhB,C;4oACI,OAAO,QAAS,WAAhB,C;8oACI,OAAO,QAAS,WAAhB,C;gpACI,OAAO,QAAS,WAAhB,C;kpACI,OAAO,QAAS,WAAhB,C;opACI,OAAO,QAAS,WAAhB,C;spACI,OAAO,QAAS,WAAhB,C;wpACI,OAAO,QAAS,WAAhB,C;0pACI,OAAO,QAAS,WAAhB,C;4pACI,OAAO,QAAS,WAAhB,C;8pACI,OAAO,QAAS,WAAhB,C;gqACI,OAAO,QAAS,WAAhB,C;kqACI,OAAO,QAAS,WAAhB,C;oqACI,OAAO,QAAS,WAAhB,C;sqACI,OAAO,QAAS,WAAhB,C;wqACI,OAAO,QAAS,WAAhB,C;0qACI,OAAO,QAAS,WAAhB,C;4qACI,OAAO,QAAS,WAAhB,C;8qACI,OAAO,QAAS,WAAhB,C;grACI,OAAO,QAAS,WAAhB,C;krACI,OAAO,QAAS,WAAhB,C;orACI,OAAO,QAAS,WAAhB,C;srACI,OAAO,QAAS,WAAhB,C;wrACI,OAAO,QAAS,WAAhB,C;0rACI,OAAO,QAAS,WAAhB,C;4rACI,OAAO,QAAS,WAAhB,C;8rACI,OAAO,QAAS,WAAhB,C;gsACI,OAAO,QAAS,WAAhB,C;ksACI,OAAO,QAAS,WAAhB,C;osACI,OAAO,QAAS,WAAhB,C;ssACI,OAAO,QAAS,WAAhB,C;wsACI,OAAO,QAAS,WAAhB,C;0sACI,OAAO,QAAS,WAAhB,C;4sACI,OAAO,QAAS,WAAhB,C;8sACI,OAAO,QAAS,WAAhB,C;gtACI,OAAO,QAAS,WAAhB,C;ktACI,OAAO,QAAS,WAAhB,C;otACI,OAAO,QAAS,WAAhB,C;stACI,OAAO,QAAS,WAAhB,C;wtACI,OAAO,QAAS,WAAhB,C;0tACI,OAAO,QAAS,WAAhB,C;4tACI,OAAO,QAAS,WAAhB,C;8tACI,OAAO,QAAS,WAAhB,C;guACI,OAAO,QAAS,WAAhB,C;kuACI,OAAO,QAAS,WAAhB,C;ouACI,OAAO,QAAS,WAAhB,C;suACI,OAAO,QAAS,WAAhB,C;wuACI,OAAO,QAAS,WAAhB,C;0uACI,OAAO,QAAS,WAAhB,C;4uACI,OAAO,QAAS,WAAhB,C;8uACI,OAAO,QAAS,WAAhB,C;gvACI,OAAO,QAAS,WAAhB,C;kvACI,OAAO,QAAS,WAAhB,C;ovACI,OAAO,QAAS,WAAhB,C;svACI,OAAO,QAAS,WAAhB,C;wvACI,OAAO,QAAS,WAAhB,C;0vACI,OAAO,QAAS,WAAhB,C;4vACI,OAAO,QAAS,WAAhB,C;8vACI,OAAO,QAAS,WAAhB,C;gwACI,OAAO,QAAS,WAAhB,C;kwACI,OAAO,QAAS,WAAhB,C;owACI,OAAO,QAAS,WAAhB,C;swACI,OAAO,QAAS,WAAhB,C;wwACI,OAAO,QAAS,WAAhB,C;0wACI,OAAO,QAAS,WAAhB,C;4wACI,OAAO,QAAS,WAAhB,C;8wACI,OAAO,QAAS,WAAhB,C;gxACI,OAAO,QAAS,WAAhB,C;kxACI,OAAO,QAAS,WAAhB,C;oxACI,OAAO,QAAS,WAAhB,C;sxACI,OAAO,QAAS,WAAhB,C;wxACI,OAAO,QAAS,WAAhB,C;0xACI,OAAO,QAAS,WAAhB,C;4xACI,OAAO,QAAS,WAAhB,C;8xACI,OAAO,QAAS,WAAhB,C;gyACI,OAAO,QAAS,WAAhB,C;kyACI,OAAO,QAAS,WAAhB,C;oyACI,OAAO,QAAS,WAAhB,C;syACI,OAAO,QAAS,WAAhB,C;wyACI,OAAO,QAAS,WAAhB,C;0yACI,OAAO,QAAS,WAAhB,C;4yACI,OAAO,QAAS,WAAhB,C;8yACI,OAAO,QAAS,WAAhB,C;gzACI,OAAO,QAAS,WAAhB,C;kzACI,OAAO,QAAS,WAAhB,C;ozACI,OAAO,QAAS,WAAhB,C;szACI,OAAO,QAAS,WAAhB,C;wzACI,OAAO,QAAS,WAAhB,C;0zACI,OAAO,QAAS,WAAhB,C;4zACI,OAAO,QAAS,WAAhB,C;8zACI,OAAO,QAAS,WAAhB,C;g0ACI,OAAO,QAAS,WAAhB,C;k0ACI,OAAO,QAAS,WAAhB,C;o0ACI,OAAO,QAAS,WAAhB,C;s0ACI,OAAO,QAAS,WAAhB,C;w0ACI,OAAO,QAAS,WAAhB,C;0;ACI,OAAO,QAAS,WAAhB,C;4;ACI,OAAO,QAAS,WAAhB,C;8;ACI,OAAO,QAAS,WAAhB,C;g<ACI,OAAO,QAAS,WAAhB,C;k<ACI,OAAO,QAAS,WAAhB,C;o<ACI,OAAO,QAAS,WAAhB,C;s<ACI,OAAO,QAAS,WAAhB,C;w<ACI,OAAO,QAAS,WAAhB,C;0<ACI,OAAO,QAAS,WAAhB,C;4<ACI,OAAO,QAAS,WAAhB,C;8<ACI,OAAO,QAAS,WAAhB,C;g=ACI,OAAO,QAAS,WAAhB,C;k=ACI,OAAO,QAAS,WAAhB,C;o=ACI,OAAO,QAAS,WAAhB,C;s=ACI,OAAO,QAAS,WAAhB,C;w=ACI,OAAO,QAAS,WAAhB,C;0=ACI,OAAO,QAAS,WAAhB,C;4=ACI,OAAO,QAAS,WAAhB,C;8=ACI,OAAO,QAAS,WAAhB,C;g>ACI,OAAO,QAAS,WAAhB,C;k>ACI,OAAO,QAAS,WAAhB,C;o>ACI,OAAO,QAAS,WAAhB,C;s>ACI,OAAO,QAAS,WAAhB,C;w>ACI,OAAO,QAAS,WAAhB,C;0>ACI,OAAO,QAAS,WAAhB,C;4>ACI,OAAO,QAAS,WAAhB,C;8>ACI,OAAO,QAAS,WAAhB,C;g?ACI,OAAO,QAAS,WAAhB,C;k?ACI,OAAO,QAAS,WAAhB,C;o?ACI,OAAO,QAAS,WAAhB,C;s?ACI,OAAO,QAAS,WAAhB,C;w?ACI,OAAO,QAAS,WAAhB,C;0?ACI,OAAO,QAAS,WAAhB,C;4?ACI,OAAO,QAAS,WAAhB,C;8?ACI,OAAO,QAAS,WAAhB,C;g@ACI,OAAO,QAAS,WAAhB,C;k@ACI,OAAO,QAAS,WAAhB,C;o@ACI,OAAO,QAAS,WAAhB,C;s@ACI,OAAO,QAAS,WAAhB,C;w@ACI,OAAO,QAAS,WAAhB,C;0@ACI,OAAO,QAAS,WAAhB,C;4@ACI,OAAO,QAAS,WAAhB,C;8@ACI,OAAO,QAAS,WAAhB,C;gAACI,OAAO,QAAS,WAAhB,C;kAACI,OAAO,QAAS,WAAhB,C;oACI,OAAO,QAAS,WAAhB,C;sACI,OAAO,QAAS,WAAhB,C;wACI,OAAO,QAAS,WAAhB,C;0ACI,OAAO,QAAS,WAAhB,C;4ACI,OAAO,QAAS,WAAhB,C;8ACI,OAAO,QAAS,WAAhB,C;gBACI,OAAO,QAAS,WAAhB,C;kBACI,OAAO,QAAS,WAAhB,C;oBACI,OAAO,QAAS,WAAhB,C;sBACI,OAAO,QAAS,WAAhB,C;wBACI,OAAO,QAAS,WAAhB,C;0BACI,OAAO,QAAS,WAAhB,C;4BACI,OAAO,QAAS,WAAhB,C;8BACI,OAAO,QAAS,WAAhB,C;gCACI,OAAO,QAAS,WAAhB,C;kCACI,OAAO,QAAS,WAAhB,C;oCACI,OAAO,QAAS,WAAhB,C;sCACI,OAAO,QAAS,WAAhB,C;wCACI,OAAO,QAAS,WAAhB,C;0CACI,OAAO,QAAS,WAAhB,C;4CACI,OAAO,QAAS,WAAhB,C;8CACI,OAAO,QAAS,WAAhB,C;gDACI,OAAO,QAAS,WAAhB,C;kDACI,OAAO,QAAS,WAAhB,C;oDACI,OAAO,QAAS,WAAhB,C;sDACI,OAAO,QAAS,WAAhB,C;wDACI,OAAO,QAAS,WAAhB,C;0DACI,OAAO,QAAS,WAAhB,C;4DACI,OAAO,QAAS,WAAhB,C;8DACI,OAAO,QAAS,WAAhB,C;gEACI,OAAO,QAAS,WAAhB,C;kEACI,OAAO,QAAS,WAAhB,C;oEACI,OAAO,QAAS,WAAhB,C;sEACI,OAAO,QAAS,WAAhB,C;wEACI,OAAO,QAAS,WAAhB,C;0EACI,OAAO,QAAS,WAAhB,C;4EACI,OAAO,QAAS,WAAhB,C;8EACI,OAAO,QAAS,WAAhB,C;gFACI,OAAO,QAAS,WAAhB,C;kFACI,OAAO,QAAS,WAAhB,C;oFACI,OAAO,QAAS,WAAhB,C;sFACI,OAAO,QAAS,WAAhB,C;wFACI,OAAO,QAAS,WAAhB,C;0FACI,OAAO,QAAS,WAAhB,C;4FACI,OAAO,QAAS,WAAhB,C;8FACI,OAAO,QAAS,WAAhB,C;gGACI,OAAO,QAAS,WAAhB,C;kGACI,OAAO,QAAS,WAAhB,C;oGACI,OAAO,QAAS,WAAhB,C;sGACI,OAAO,QAAS,WAAhB,C;wGACI,OAAO,QAAS,WAAhB,C;0GACI,OAAO,QAAS,WAAhB,C;4GACI,OAAO,QAAS,WAAhB,C;8GACI,OAAO,QAAS,WAAhB,C;gHACI,OAAO,QAAS,WAAhB,C;kHACI,OAAO,QAAS,WAAhB,C;oHACI,OAAO,QAAS,WAAhB,C;sHACI,OAAO,QAAS,WAAhB,C;wHACI,OAAO,QAAS,WAAhB,C;0HACI,OAAO,QAAS,WAAhB,C;4HACI,OAAO,QAAS,WAAhB,C;8HACI,OAAO,QAAS,WAAhB,C;gIACI,OAAO,QAAS,WAAhB,C;kIACI,OAAO,QAAS,WAAhB,C;oIACI,OAAO,QAAS,WAAhB,C;sIACI,OAAO,QAAS,WAAhB,C;wIACI,OAAO,QAAS,WAAhB,C;0IACI,OAAO,QAAS,WAAhB,C;4IACI,OAAO,QAAS,WAAhB,C;8IACI,OAAO,QAAS,WAAhB,C;gJACI,OAAO,QAAS,WAAhB,C;kJACI,OAAO,QAAS,WAAhB,C;oJACI,OAAO,QAAS,WAAhB,C;sJACI,OAAO,QAAS,WAAhB,C;wJACI,OAAO,QAAS,WAAhB,C;0JACI,OAAO,QAAS,WAAhB,C;4JACI,OAAO,QAAS,WAAhB,C;8JACI,OAAO,QAAS,WAAhB,C;gKACI,OAAO,QAAS,WAAhB,C;kKACI,OAAO,QAAS,WAAhB,C;oKACI,OAAO,QAAS,WAAhB,C;sKACI,OAAO,QAAS,WAAhB,C;wKACI,OAAO,QAAS,WAAhB,C;0KACI,OAAO,QAAS,WAAhB,C;4KACI,OAAO,QAAS,WAAhB,C;8KACI,OAAO,QAAS,WAAhB,C;gLACI,OAAO,QAAS,WAAhB,C;kLACI,OAAO,QAAS,WAAhB,C;oLACI,OAAO,QAAS,WAAhB,C;sLACI,OAAO,QAAS,WAAhB,C;wLACI,OAAO,QAAS,WAAhB,C;0LACI,OAAO,QAAS,WAAhB,C;4LACI,OAAO,QAAS,WAAhB,C;8LACI,OAAO,QAAS,WAAhB,C;gMACI,OAAO,QAAS,WAAhB,C;kMACI,OAAO,QAAS,WAAhB,C;oMACI,OAAO,QAAS,WAAhB,C;sMACI,OAAO,QAAS,WAAhB,C;wMACI,OAAO,QAAS,WAAhB,C;0MACI,OAAO,QAAS,WAAhB,C;4MACI,OAAO,QAAS,WAAhB,C;8MACI,OAAO,QAAS,WAAhB,C;gNACI,OAAO,QAAS,WAAhB,C;kNACI,OAAO,QAAS,WAAhB,C;oNACI,OAAO,QAAS,WAAhB,C;sNACI,OAAO,QAAS,WAAhB,C;wNACI,OAAO,QAAS,WAAhB,C;0NACI,OAAO,QAAS,WAAhB,C;4NACI,OAAO,QAAS,WAAhB,C;8NACI,OAAO,QAAS,WAAhB,C;gOACI,OAAO,QAAS,WAAhB,C;kOACI,OAAO,QAAS,WAAhB,C;oOACI,OAAO,QAAS,WAAhB,C;sOACI,OAAO,QAAS,WAAhB,C;wOACI,OAAO,QAAS,WAAhB,C;0OACI,OAAO,QAAS,WAAhB,C;4OACI,OAAO,QAAS,WAAhB,C;8OACI,OAAO,QAAS,WAAhB,C;gPACI,OAAO,QAAS,WAAhB,C;kPACI,OAAO,QAAS,WAAhB,C;oPACI,OAAO,QAAS,WAAhB,C;sPACI,OAAO,QAAS,WAAhB,C;wPACI,OAAO,QAAS,WAAhB,C;0PACI,OAAO,QAAS,WAAhB,C;4PACI,OAAO,QAAS,WAAhB,C;8PACI,OAAO,QAAS,WAAhB,C;gQACI,OAAO,QAAS,WAAhB,C;kQACI,OAAO,QAAS,WAAhB,C;oQACI,OAAO,QAAS,WAAhB,C;sQACI,OAAO,QAAS,WAAhB,C;wQACI,OAAO,QAAS,WAAhB,C;0QACI,OAAO,QAAS,WAAhB,C;4QACI,OAAO,QAAS,WAAhB,C;8QACI,OAAO,QAAS,WAAhB,C;gRACI,OAAO,QAAS,WAAhB,C;kRACI,OAAO,QAAS,WAAhB,C;oRACI,OAAO,QAAS,WAAhB,C;sRACI,OAAO,QAAS,WAAhB,C;wRACI,OAAO,QAAS,WAAhB,C;0RACI,OAAO,QAAS,WAAhB,C;4RACI,OAAO,QAAS,WAAhB,C;8RACI,OAAO,QAAS,WAAhB,C;gSACI,OAAO,QAAS,WAAhB,C;kSACI,OAAO,QAAS,WAAhB,C;oSACI,OAAO,QAAS,WAAhB,C;sSACI,OAAO,QAAS,WAAhB,C;wSACI,OAAO,QAAS,WAAhB,C;0SACI,OAAO,QAAS,WAAhB,C;4SACI,OAAO,QAAS,WAAhB,C;8SACI,OAAO,QAAS,WAAhB,C;gTACI,OAAO,QAAS,WAAhB,C;kTACI,OAAO,QAAS,WAAhB,C;oTACI,OAAO,QAAS,WAAhB,C;sTACI,OAAO,QAAS,WAAhB,C;wTACI,OAAO,QAAS,WAAhB,C;0TACI,OAAO,QAAS,WAAhB,C;4TACI,OAAO,QAAS,WAAhB,C;8TACI,OAAO,QAAS,WAAhB,C;gUACI,OAAO,QAAS,WAAhB,C;kUACI,OAAO,QAAS,WAAhB,C;oUACI,OAAO,QAAS,WAAhB,C;sUACI,OAAO,QAAS,WAAhB,C;wUACI,OAAO,QAAS,WAAhB,C;0UACI,OAAO,QAAS,WAAhB,C;4UACI,OAAO,QAAS,WAAhB,C;8UACI,OAAO,QAAS,WAAhB,C;gVACI,OAAO,QAAS,WAAhB,C;kVACI,OAAO,QAAS,WAAhB,C;oVACI,OAAO,QAAS,WAAhB,C;sVACI,OAAO,QAAS,WAAhB,C;wVACI,OAAO,QAAS,WAAhB,C;0VACI,OAAO,QAAS,WAAhB,C;4VACI,OAAO,QAAS,WAAhB,C;8VACI,OAAO,QAAS,WAAhB,C;gWACI,OAAO,QAAS,WAAhB,C;kWACI,OAAO,QAAS,WAAhB,C;oWACI,OAAO,QAAS,WAAhB,C;sWACI,OAAO,QAAS,WAAhB,C;wWACI,OAAO,QAAS,WAAhB,C;0WACI,OAAO,QAAS,WAAhB,C;4WACI,OAAO,QAAS,WAAhB,C;8WACI,OAAO,QAAS,WAAhB,C;gXACI,OAAO,QAAS,WAAhB,C;kXACI,OAAO,QAAS,WAAhB,C;oXACI,OAAO,QAAS,WAAhB,C;sXACI,OAAO,QAAS,WAAhB,C;wXACI,OAAO,QAAS,WAAhB,C;0XACI,OAAO,QAAS,WAAhB,C;4XACI,OAAO,QAAS,WAAhB,C;8XACI,OAAO,QAAS,WAAhB,C;gYACI,OAAO,QAAS,WAAhB,C;kYACI,OAAO,QAAS,WAAhB,C;oYACI,OAAO,QAAS,WAAhB,C;sYACI,OAAO,QAAS,WAAhB,C;wYACI,OAAO,QAAS,WAAhB,C;0YACI,OAAO,QAAS,WAAhB,C;4YACI,OAAO,QAAS,WAAhB,C;8YACI,OAAO,QAAS,WAAhB,C;gZACI,OAAO,QAAS,WAAhB,C;kZACI,OAAO,QAAS,WAAhB,C;oZACI,OAAO,QAAS,WAAhB,C;sZACI,OAAO,QAAS,WAAhB,C;wZACI,OAAO,QAAS,WAAhB,C;0ZACI,OAAO,QAA

AAI,8BAAJ,C;UACI,IAAI,sCAAJ,C;YAC0B,qB;YAAtB,iBAAc,CAAd,wB;cACI,IAAK,WAAI,sBAAK,KAAL,CAAJ,C;;YAEI,wCAAa,CAAb,C;YAAb,OAAa,gBAAb,C;cAAa,wB;cACT,IAAK,WAAI,IAAJ,C;;;UAEb,OAAO,I;;;QAIX,OAAO,gB;;MAEX,YAAY,C;MACC,6B;MAAb,OAAa,gBAAb,C;QAAa,0B;QACT,IAAI,SAAS,CAAb,C;UAAgB,IAAK,WAAI,MAAJ,C;;UAAe,qB;;MAExC,OAAY,qBAAL,IAAK,C;K;IAGhB,kC;MNnqBI,IAAI,EM2qBI,KAAK,CN3qBT,CAAJ,C;QACI,cM0qBc,sD;QNzqBd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MM0qBV,OAAO,kBAAgB,gBAAO,IAAO,OAAP,CAAhB,CAAP,CAAO,EAAE,EAAc,CAAd,EAAiB,CAAjB,C;K;kGAAT,yB;MAAA,4C;MAAA,qD;MAAA,uC;QAAA,eAAA,+BAAA,cAAA,C;UAAA,eAAA,OAAA,QAAA,cAAA,C;YAAA,IAAI,CAAC,mBAAL,C;UAAA,eAAA,OAAA,QAAA,cAAA,C;YAAA,IAAI,CAAC,UAAU,QAAf,C;;;

;UACJ,IAAK,WAAI,IAAJ,C;;QAET,OAAO,I;O;KAZX,C;IAoBA,+B;MAII,IAAI,wCAAsB,kBAAQ,CAAlC,C;QAAqC,OAAO,mB;MAC5C,WAAW,0B;MACN,WAAL,IAAK,C;MACL,OAAO,I;K;IAGX,uC;MAOI,aAAU,2BAAV,OAA2B,CAA3B,M;QACI,QAAQ,MAAO,iBAAQ,IAAI,CAAJ,IAAR,C;QACf,sBAAK,CAAL,EAAU,SAAK,aAAI,CAAJ,EAAO,CAAP,EAAU,CAAV,C;;K;oFAIR,yB;MAAA,oD;MJn4BA,sC;MAAA,oC;MAAA,uBAOe,yB;QAArE,8D;QAAA,4B;QAAA,4B;QAAA,uBAAqE,gB;QAAA,yB;QAAA,yB;QAAA,uBAAA

AAgB,cAAhB,C;QAAgB,yB;QACZ,WAAe,UAAU,OAAV,C;QbpkBnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OA
ArB,C;;MaskBA,OAAO,W;K;kGAGX,yB;MAAA,kF;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,2C;QAYI,aAAa,
mBAA6D,cAAzC,YAAY,mCAAwB,EAAxB,CAAZ,CAAyC,EAAc,EAAd,CAA7D,C;QAcG,Q;QAAA,2B;QAAh
B,OAAgB,cAAhB,C;UAAgB,yB;UAbO,MAcP,aAAI,OAAJ,EAde,aAcF,CAAc,OAAd,CAAb,C;;QAdhB,OAAuB,
M;O;KAb3B,C;sGAgBA,iD;MAUoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,WAAY,aAAI,
OAAJ,EAAa,cAAc,OAAd,CAAb,C;;MAEhB,OAAO,W;K;IAGX,gD;MAIiB,Q;MAAA,2B;MAAb,OAAa,cAAb,C;
QAAa,sB;QACT,WAAY,WAAI,IAAJ,C;;MAEhB,OAAO,W;K;IAGX,gC;MAII,OAAO,0BAAa,eAAW,YAAY,m
CAAwB,EAAxB,CAAZ,CAAyC,EAAc,EAAd,CAAb,C;K;IAGX,6B;MAKqB,GAAkB,sBAAI,CAAJ,C;QACW,QAAM,c
AAN,C;eACH,C;YAAK,kB;YAAL,K;eACA,C;YAAK,cAAW,8BAAJ,GAAkB,sBAAI,CAAJ,CAAlB,GAA8B,oB
AAW,OAAhD,C;YAAL,K;;YACa,uBAAL,SAAK,C;YAAV,K;;QAAP,W;;MAMJ,OAA4B,qBAAhB,hB,gBAAL,SA
AK,CAAgB,C;K;IAGhC,oC;MAII,IAAI,oCAAJ,C;QACI,OAAY,gBAAL,SAAK,C;MAChB,OAAO,0BAAa,gBA
Ab,C;K;IAGX,oC;MAII,OAAO,iBAAU,SAAV,C;K;IAGX,4B;MAOqB,GAAkB,sBAAI,CAAJ,C;QACW,QAAM,Q
AAM,cAAN,C;eACH,C;YAAK,iB;YAAL,K;eACA,C;YAAK,aAAU,8BAAJ,GAAkB,sBAAK,CAAL,CAAlB,CAAA+B,GA
A+B,oBAAW,OAAhD,C;YAAL,K;;YACQ,iCAAa,qBAAiB,YAAY,cAAZ,CAAjB,CAAb,C;YAHL,K;;QAAP,W;;
MAMJ,OAAwC,oBAAjC,0BAAa,sBAAb,CAAiC,C;K;sFAG5C,yB;MAAA,+D;MAwFA,gD;MAxFA,uC;QAMW,
kBAAU,gB;QAsFD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAvF6B,SAuFlB,CAAU,OAAA
V,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxFhB,OA0FO,W;O;KAhGX,C;uFASA,yB;MAAA,+D;MA0FA,gD;MA1
FA,uC;QAMW,kBAAU,gB;QAwFD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAzF6B,SAyFlB,
B,SAyFlB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1FhB,OA4FO,W;O;KAtGX,C;oGAaA,yB;MAAA,
MAAA,+D;MA8BA,gD;MA9BA,uC;QAOW,kBAAU,iB;QA6BD,gB;QA6BR,gB;QADhB,YAAY,C;QACI,2B;Q
AAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA9BoC,SA8BzB,CAAU,oBAAmB,cAAnB,EAAmB,sBAAnB,UA
AV,EAAuC,OAAvC,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/BhB,OAiCO,W;O;KA7CX,C;oGAaA,yB;MAAA,
MAAA,+D;MAiCA,gD;MAjCA,uC;QAOW,kBAAU,iB;QAgCD,gB;QAgCR,gB;QADhB,YAAY,C;QACI,2B;QAAh
B,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAjCoC,SAiCzB,CAAU,oBAAmB,cAAnB,EAAmB,sBAAnB,UAAV,E
AAuC,OAAvC,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlChB,OAoCO,W;O;KAhDX,C;oGAaA,yB;MAAA,
wE;MAAA,gD;MAAA,oD;QAAA,oB,UAC4B,M;QAAF,5C,YAAA,Y,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,
yB;UACZ,WAAW,UAAU,oBAAmB,cAAnB,EAAmB,sBAAnB,UAAV,EAAuC,OAAvC,C;UACC,OAAZ,WAAY
,EAAO,IAAP,C;;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAW,UAAU,oBAAmB,cAAnB,EAAmB,sBAAnB,UAAV,E
AAuC,OAAvC,C;UACC,OAAZ,WAAY

C;;QAET,OAAO,W;O;KAfX,C;4FAkBA,yB;MAAA,kC;MAAA,4C;MAAA,wE;QAQW,sC;QAAA,8C;O;MARX,oDASQ,Y;QAA6C,OAAA,oBAAgB,W;O;MATrE,iDAUQ,mB;QAAoC,gCAAY,OAAZ,C;O;MAV5C,gF;MAAA,yC;QAQI,2D;O;KARJ,C;8EAcA,yB;MAAA,kF;MAAA,gE;MAAA,uC;QAOW,kBAAM,eAAa,mCAAwB,EAAxB,CAAb,C;QAuEA,Q;QAAA,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAAxEwC,SAAwEpC,CAAAU,IAAAV,CAAAJ,C;;QAAxEhB,OAAyEO,W;O;KAAhFX,C;4FAAUA,yB;MAAA,kF;MAAA,gE;MA+BA,uC;QAAOW,kBAAMa,eAAa,mCAAwB,EAAxB,CAAb,C;QAAgCP,gB;QAAbD,Y;QAAAA,C;QAAAC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAAjC+C,SAAiCC,CAAAC,IAAAD,CAAAE,C;;QAAjCC,OAAkCC,W;O;KAAzCC,C;8GAAUA,yB;MAAA,+D;MAAA,+E;MAAA,uC;QAAOC,kBAAMC,eAAa,mCAAwB,EAAxB,CAAb,C;QAAgCC,gB;QAAbC,Y;QAAAA,C;QAAAC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAAjCC

AGW,sB;;QAUP,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eAdmB,QAcJ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QAjBe,QAiBP,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MAvBP,yB;K;8FAGJ,+B;MAOI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,cAAc,QAAS,O;MACvB,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,O;MAChC,eAAe,SAAS,OAAT,C;;QAEX,QAAQ,QAAS,O;QACjB,QAAQ,SAAS,CAAT,C;QACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;MAED,QAAT,QAAS,W;MAClB,OAAO,O;K;mFAGX,yB;MAAA,sE;MF/yDA,iB;ME+yDA,sC;QAaI,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,yB;UAAA,6B;QAAA,C;QAAA,sC;UAAA,qB;QAAA,C;QAAA,sC;UAAA,qB;QAAA,C;QAAA,sC;UAAA,qB

I,CAAC,QAAS,UAAd,C;UAAyB,MAAM,6B;QAC/B,eAAe,SAAS,QAAS,OAAlB,C;QACf,OAAO,QAAS,UAAh
B,C;UACI,QAAQ,SAAS,QAAS,OAAlB,C;UACR,WF54DG,MAAO,KE44DO,QF54DP,EE44DiB,CF54DjB,C;;Q
E84Dd,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MFp6DA,iB;MEo6DA,sC;QAaI,eAAe,oB;QACf,IAAI,C
AAC,QAAS,UAAd,C;UAAyB,MAAM,6B;QAC/B,eAAe,SAAS,QAAS,OAAlB,C;QACf,OAAO,QAAS,UAAhB,C
;UACI,QAAQ,SAAS,QAAS,OAAlB,C;UACR,WF96DG,MAAO,KE86DO,QF96DP,EE86DiB,CF96DjB,C;;QEg7
Dd,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAAA,sC;QAWI,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAd
,C;UAAyB,MAAM,6B;QAC/B,eAAe,SAAS,QAAS,OAAlB,C;QACf,OAAO,QAAS,UAAhB,C;UACI,QAAQ,SA
AS,QAAS,OAAlB,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;+
FAuBA,yB;MFv8DA,iB;MEu8DA,sC;QAWI,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,OAAO,I;QA
ChC,eAAe,SAAS,QAAS,OAAlB,C;QACf,OAAO,QAAS,UAAhB,C;UACI,QAAQ,SAAS,QAAS,OAAlB,C;UACR
,WF/8DG,MAAO,KE+8DO,QF/8DP,EE+8DiB,CF/8DjB,C;;QEi9Dd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MFv+D
A,iB;MEu+DA,sC;QAWI,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,OAAO,I;QAChC,eAAe,SAAS,Q
AAS,QAAS,OAAlB,C;QACf,OAAO,QAAS,UAAhB,C;UACI,QAAQ,SAAS,QAAS,OAAlB,C;UACR,WF/+DG,MAAO,
KE++DO,QF/+DP,EE++DiB,CF/+DjB,C;;QEi/Dd,OAAO,Q;O;KAlBX,C;+FAqBA,+B;MASI,eAAe,oB;MACf,IA
AI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,eAAe,SAAS,QAAS,OAAlB,C;MACf,OAAO,QAAS,UAAh
B,C;QACI,QAAQ,SAAS,QAAS,OAAlB,C;QACR,IAAI,2BAAW,CAAX,KAAJ,C;UACI,WAAW,C;;;MAGnB,OA
AO,Q;K;0FAGX,yB;MAAA,sE;MAAA,kD;QAWI,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,MAAM
,6B;QAC/B,eAAe,SAAS,QAAS,OAAlB,C;QACf,OAAO,QAAS,UAAhB,C;UACI,QAAQ,SAAS,QAAS,OAAlB,C
;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAA
O,Q;O;KApBX,C;sGAuBA,2C;MASI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,e
AAe,SAAS,QAAS,OAAlB,C;MACf,OAAO,QAAS,UAAhB,C;QACI,QAAQ,SAAS,QAAS,OAAlB,C;QACR,IAAI
,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;UACI,WAAW,C;;;MAGnB,OAAO,Q;K;IAGX,
gC;MAOI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OA
AO,QAAS,UAAhB,C;QACI,QAAQ,SAAS,O;QACjB,MFtjEG,MAAO,KEsjEE,GFtjEF,EEsjEO,CFtjEP,C;;MEwj
Ed,OAAO,G;K;IAGX,iC;MAOI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,UAAU,
QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,SAAS,O;QACjB,MFllEG,MAAO,KEklEE,GFllEF,EE
klEO,CFllEP,C;;MEolEd,OAAO,G;K;IAGX,iC;MAKI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OA
AO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,SAAS,O;QACjB,IAAI,sBAAM,
CAAN,KAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,0C;MAGI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,
C;QAAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,SAAS,O;QACjB,IAAI
,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;UACI,WAAW,C;;;MAGnB,OAAO,Q;K;IAGX,gD;
MAKI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,
QAAS,UAAhB,C;QACI,QAAQ,SAAS,O;QACjB,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC
,C;UACI,WAAW,C;;;MAGnB,OAAO,Q;K;IAGX,gC;MAOI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAy
B,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,SAAS,O;QACjB,IAAI,UAAW,
SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;UACI,WAAW,C;;;MAGnB,OAAO,Q;K;IAGX,gC;MAOI,eAAe,
oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;
QACI,QAAQ,SAAS,O;QACjB,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;UACI,WAAW,C;
;;MAGnB,OAAO,Q;K;IAGX,gC;MAOI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,UAAU
,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,SAAS,O;QACjB,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,
CAAX,GAAkC,CAAtC,C;UACI,WAAW,C;;;MAGnB,OAAO,Q;K;IAGX,gC;MAOI,eAAe,oB;MACf,IAAI,CAAC,QAAS,
UAAd,C;QAAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,SAAS,O;QACjB,IAAI
,UAAW,

mB,oBAAnB,QAAV,EAAuC,WAAvC,EAAoD,QAAS,OAA7D,C;;QAElB,OAAO,W;O;KApBX,C;gGAuBA,gC;MAcI,eAAe,SAAK,W;MACpB,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,kBAAqB,QAAS,O;MAC9B,OAAO,QAAS,UAAhB,C;QACI,cAAc,UAAU,WAAV,EAAuB,QAAS,OAAhC,C;;MAElB,OAAO,W;K;8FAGX,yB;MAAA,4F;MAAA,uC;QAAaI,eAAe,+BAAa,cAAAb,C;QACf,IAAI,CAAC,QAAS,cAAd,C;QACI,IAAI,CAAC,QAAS,gB;UACrB,cAAc,UAAU,KAAV,EAAiB,QAAS,WAA1B,B,EAAsC,WAAtC,C;;QAElB,OAAO,W;O;KArBX,C;wHAwBA,gC;MAAaI,eAAe,+BAAa,cAAAb,C;MACf,IAAI,CAAC,QAAS,cAAd,C;MACI,IAAI,CAAC,QAAS,I;QACrB,MAAM,kBAAqB,QAAS,W;QAC9B,OAAO,QAAS,cAAhB,C;QACI,YAAY,QAAS,gB;UACrB,cAAc,UAAU,KAAV,EAAiB,QAAS,WAA1B,B,EAAsC,WAAtC,C;;MAElB,OAAO,W;K;0FAGX,gC;MAcI,eAAe,+BAAa,cAAAb,C;MACf,IAAI,CAAC,QAAS,cAAd,C;MACI,IAAI,CAAC,QAAS,I;QACrB,MAAM,kBAAqB,QAAS,W;QAC9B,OAAO,QAAS,cAAhB,C;QACI,cAAc,UAAU,QAAV,EAAoB,QAAS,WAA7B,B,EAAyC,WAAzC,C;;MAElB,OAAO,W;K;8FAGX,kB;MAAA,kF;MAAA,gD;MAAA,gE;MAAA,gD;QAAiBI,oBAAoB,QAAS,oBAAT,C;QACpB,IAAI,kBAAqB,cAAAb,C;QACzB,IAAI,CAAC,QAAS,cAAd,C;QACI,IAAI,CAAC,QAAS,I;UACrB,MAAM,kBAAqB,QAAS,W;UAC9B,OAAO,QAAS,cAAhB,C;UACI,YAAY,QAAS,gB;YACrB,cAAc,UAAU,KAAV,EAAiB,QAAS,WAA1B,B,EAAsC,WAAtC,C;;QAElB,OAAO,W;O;KAtBX,C;4GAyBA,gC;MAAaI,eAAe,+BAAa,cAAAb,C;MACf,IAAI,CAAC,QAAS,cAAd,C;MACI,IAAI,CAAC,QAAS,I;QACrB,MAAM,kBAAqB,QAAS,W;QAC9B,OAAO,QAAS,cAAhB,C;QACI,YAAY,QAAS,gB;UACrB,cAAc,UAAU,KAAV,EAAiB,QAAS,WAA1B,B,EAAsC,WAAt

AAS,OAAT,CjBnwFoE,KAAX,IAAf,C;;QiBqwFrD,OAAO,G;O;KAdX,C;mFAiBA,yB;MD78EA,+B;MC68EA,sC;QAWoB,Q;QADhB,UD58EqC,eAAW,oBC48E/B,CD58E+B,CAAX,C;QC68ErB,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,MDjxFmD,eCixFnD,GDjxF8D,KAAK,KCixF5D,SAAS,OAAT,CDjxFuE,KAAX,CAAhB,C;;QCmxFvD,OAAO,G;O;KAdX,C;IAiBA,qC;MAIoB,UAMT,M;MANS,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MAAI,eAAJ,C;UACI,MAAM,gCAAyB,2BAAwB,SAAxB,MAAwB,zB,C;;;MAId,OAAO,mE;K;IAGX,qC;MAIoB,UAMT,M;MANS,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MAAI,eAAJ,C;UACI,MAAM,gCAAyB,2BAAwB,SAAxB,MAAwB,zB,C;;;MAId,OAAO,+D;K;IAGX,sC;MAIoB,UAMT,M;MANS,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MAAI,eAAJ,C;UACI,MAAM,gCAAyB,2BAAwB,SAAxB,MAAwB,zB,C;;;MAId,OAAO,mE;K;IAGX,mC;MAAI,aAAa,iBAAa,mCAAwB,EAAxB,sB,CAAwB,C;MAAC,kBAAc,KAAd,C;MAAnlEgB,Q;MAAA,OAAOlET,SAAplES,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MAAI,eAAJ,C;UACI,MAAM,gCAAyB,2BAAwB,SAAxB,MAAwB,zB,C;;;MAId,OAAO,mE;K;IAGX,qC;MAIoB,UAMT,M;MANS,2B

CA,IAAI,0CAAwB,8BAA5B,C;QACI,eAAe,SAAK,K;QACpB,qBAAqB,YAAW,IAAX,SAAsB,WAAW,IAAX,K
AAmB,CAAvB,GAA0B,CAA1B,GAAiC,CAAnD,K;QACrB,aAAa,iBAAa,cAAb,C;QACb,eAAa,kBAAc,SAAd,C;
QACb,YAAY,C;QACZ,OAAgB,CAAT,qBAAiB,QAAxB,C;UACI,iBAAsB,eAAL,IAAK,EAAa,WAAW,KAAX,I
AAb,C;UACtB,IAAI,CAAC,cAAD,IAAmB,aAAa,IAApC,C;YA0C,K;UAC1C,QAAO,cAAK,KAAL,EAAY,QA
AQ,UAAR,IAAZ,C;UACP,MAAO,WAAI,UAAU,QAAV,CAAJ,C;UACP,gBAAS,IAAT,I;;QAEJ,OAAO,M;;MA
EX,eAAa,gB;MACgE,kBAA7E,iBAAiB,oBAAjB,EAA6B,IAA7B,EAAmC,IAAnC,EAAyC,cAAzC,EAAuE,IAAv
E,C;MEtoGA,OAAgB,qBAAhB,C;QAAgB,gC;QFuoGL,mBAAI,UEvoGiB,OFuoGjB,CAAJ,C;;MAEX,OAAO,Q;
K;IAGX,kC;MAqBoB,gB;MAHhB,gBAXW,KAWW,O;MACtB,WAAW,iBF17FJ,MAAO,KE07FgB,mCAAwB,E
AAxB,CF17FhB,EE07F6C,SF17F7C,CE07FH,C;MACX,QAAQ,C;MACQ,2B;MAAhB,OAAgB,cAAhB,C;QAAg
B,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAAhBqB,GAAgBP,OAAhBO,EAAnB,KAAgBqB,QAAhB,CAAA
M,UAAN,EAAM,kBAAN,SAAhBF,CAAgBG,EAAhB,C;;MAAhBT,TAAgBU,OAAhBV,I;K;+EAfX,yB;MAAA,kF;MAAA,gE;MFv7FA,i
B;MEu7FA,8C;QAWoB,UAEsB,M;QALtC,gBAAgB,KAAM,O;QACtB,WAAW,eF17FJ,MAAO,KE07FgB,mCA
AwB,EAAxB,CF17FhB,EE07F6C,SF17F7C,CE07FH,C;QACX,QAAQ,C;QACQ,2B;QAAhB,OAAgB,cAAhB,C;QAAg
B,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAhBqB,GAAgBP,OAAhBO,EAAnB,KAAgBqB,QAAhB,CAAA
M,UAAN,EAAM,kBAAN,SAAhBF,CAAgBG,EAAhB,C;;MAAhBT,TAAgBU,OAAhBV,I;K;+EAdX,yB;MAAA,kF;MAAA,gE;MFn
9FA,iB;MEm9FA,8C;QAQI,YAAY,oB;QACZ,aAAa,KAAM,W;QACnB,WAAW,eFv9FJ,MAAO,KEu9FgB,mCA
AwB,EAAxB,CFv9FhB,EEu9FmD,wBAAbtD,KAAasD,EA
AwB,EAAxB,CFv9FnD,CEu9FH,C;MACX,OAAO,KAAM,UAAN,IAAmBC,MAAO,UAAjC,C;QACI,IAAK,WAAfqB,
B,GAAeP,KAAM,OAAfC,EAAeO,MAAO,OAAfd,CAAeR,B,C;;MAAf

GAAgC,MAAM,K;K;IAGjD,+B;MAOoB,Q;MAFhB,UAAkB,G;MAClB,YAAiB,C;MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;QACP,oBAAmB,qBAAnB,EAAmB,KAAnB,E;;MAEJ,OAAW,UAAS,CAAb,GAAgB,wCAAO,IAAvB,GAAgC,MAAM,K;K;IAGjD,2B;MAMoB,Q;MADhB,UAAe,C;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,YAAO,O;;MAEX,OAAO,G;K;IAGX,2B;MAMoB,Q;MADhB,UAAe,C;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,YAAO,O;;MAEX,OAAO,G;K;IAGX,2B;MAMoB,Q;MADhB,UAAe,C;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,YAAO,OAAP,I;;MAEJ,OAAO,G;K;IAGX,2B;MAMoB,Q;MADhB,Y;MACgB,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,cAAO,OAAP,C;;MAEJ,OAAO,G;K;IAGX,2B;MAMoB,Q;MADhB,UAAiB,G;MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;;MAEX,OAAO,G;K;IAGX,2B;MAMoB,Q;MADhB,UAAkB,G;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;;MAEX,OAAO,G;K;IGn1GX,uC;MAOI,OAAO,SAAM,CAAN,EAAS,SAAM,CAAN,EAAS,CAAT,EAAY,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,CAA9H,EAAiI,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,CAA9H,EAAiI,CAAjI,EAAoI,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,CAA9H,EAAiI,CAAjI,EAAoI,CAApI,EAAuI,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,CAA9H,EAAiI,CAAjI,EAAoI,CAApI,EAAuI,CAAvI,EAA0I,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,CAA9H,EAAiI,CAAjI,EAAoI,CAApI,EAAuI,CAAvI,EAA0I,CAA1I,EAA6I,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,CAA9H,EAAiI,CAAjI,EAAoI,CAApI,EAAuI,CAAvI,EAA0I,CAA1I,EAA6I,CAA7I,EAAgJ,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,CAA9H,EAAiI,CAAjI,EAAoI,CAApI,EAAuI,CAAvI,EAA0I,CAA1I,EAA6I,CAA7I,EAAgJ,CAAhJ,EAAmJ,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,CAA9H,EAAiI,CAAjI,EAAoI,CAApI,EAAuI,CAAvI,EAA0I,CAA1I,EAA6I,CAA7I,EAAgJ,CAAhJ,EAAmJ,CAAnJ,EAAsJ,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,CAA9H,EAAiI,CAAjI,EAAoI,CAApI,EAAuI,CAAvI,EAA0I,CAA1I,EAA6I,CAA7I,EAAgJ,CAAhJ,EAAmJ,CAAnJ,EAAsJ,CAAtJ,EAAyJ,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B,EAA8B,CAA9B,EAAiC,CAAjC,EAAoC,CAApC,EAAuC,CAAvC,EAA0C,CAA1C,EAA6C,CAA7C,EAAgD,CAAhD,EAAmD,CAAnD,EAAsD,CAAtD,EAAyD,CAAzD,EAA4D,CAA5D,EAA+D,CAA/D,EAAkE,CAAlE,EAAqE,CAArE,EAAwE,CAAxE,EAA2E,CAA3E,EAA8E,CAA9E,EAAiF,CAAjF,EAAoF,CAApF,EAAuF,CAAvF,EAA0F,CAA1F,EAA6F,CAA7F,EAAgG,CAAhG,EAAmG,CAAnG,EAAsG,CAAtG,EAAyG,CAAzG,EAA4G,CAA5G,EAA+G,CAA/G,EAAkH,CAAlH,EAAqH,CAArH,EAAwH,CAAxH,EAA2H,CAA3H,EAA8H,CAA9H,EAAiI,CAAjI,EAAoI,CAApI,EAAuI,CAAvI,EAA0I,CAA1I,EAA6I,CAA7I,EAAgJ,CAAhJ,EAAmJ,CAAnJ,EAAsJ,CAAtJ,EAAyJ,CAAzJ,EAA4J,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,EAAe,CAAf,EAAkB,CAAlB,EAAqB,CAArB,EAAwB,CAAxB,EAA2B,CAA3B

MAMoB,Q;MAFhB,IAAI,mBAAJ,C;QAAe,OAAO,C;MACtB,YAAY,C;MACI,OAAA,SjB2KoE,QAAQ,W;MiB3K5F,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,qB;;MAC9C,OAAO,K;K;sFAGX,6B;MAKoB,Q;MAAA,OAAA,SjBkoE,QAAQ,W;MiBlK5F,OAAgB,cAAhB,C;QAAgB,yB;QAAM,OAAO,OAAP,C;;K;kFAG1B,+B;MAemB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIjpDmB,QJipDJ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIppDe,QJopDP,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI1pDP,yB;K;8FAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eItoD2B,QJsoDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIzoDuB,QJyoDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI/oDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIxoD2B,QJwoDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI3oDuB,QJ2oDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIjpDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI1oD2B,QJ0oDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI7oDuB,QJ6oDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MInpDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI5oD2B,QJ4oDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI/oDuB,QJ+oDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIrpDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI9oD2B,QJ8oDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIjpDuB,QJipDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIvpDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIhpD2B,QJgpDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QInpDuB,QJmpDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIzpDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIlpD2B,QJkpDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIrpDuB,QJqpDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI3pDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIppD2B,QJopDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIvpDuB,QJupDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI7pDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eItpD2B,QJspDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIzpDuB,QJypDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI/pDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIxpD2B,QJwpDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI3pDuB,QJ2pDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIjqDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI1pD2B,QJ0pDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI7pDuB,QJ6pDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MInqDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI5pD2B,QJ4pDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI/pDuB,QJ+pDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIrqDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI9pD2B,QJ8pDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIjqDuB,QJiqDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIvqDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIhqD2B,QJgqDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QInqDuB,QJmqDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIzqDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIlqD2B,QJkqDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIrqDuB,QJqqDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI3qDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIpqD2B,QJoqDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIvqDuB,QJuqDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI7qDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eItqD2B,QJsqDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIzqDuB,QJyqDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI/qDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIxqD2B,QJwqDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI3qDuB,QJ2qDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIjrDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI5qD2B,QJ4qDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI/qDuB,QJ+qDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIrrDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIhrD2B,QJgrDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QInrDuB,QJmrDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIzrDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIlrD2B,QJkrDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIrrDuB,QJqrDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI3rDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIprD2B,QJorDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIvrDuB,QJurDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI7rDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eItrD2B,QJsrDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIzrDuB,QJyrDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI/rDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIxrD2B,QJwrDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI3rDuB,QJ2rDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIjsDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI5rD2B,QJ4rDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI/rDuB,QJ+rDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIrsDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIhsD2B,QJgsDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QInsDuB,QJmsDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIzsDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIlsD2B,QJksDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIrsDuB,QJqsDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI3sDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIpsD2B,QJosDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIvsDuB,QJusDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI7sDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eItsD2B,QJssDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIzsDuB,QJysDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI/sDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIxsD2B,QJwsDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI3sDuB,QJ2sDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIjtDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI5sD2B,QJ4sDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI/sDuB,QJ+sDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIrtDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIhtD2B,QJgtDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIntDuB,QJmtDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIztDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIltD2B,QJktDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIrtDuB,QJqtDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI3tDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIptD2B,QJotDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIvtDuB,QJutDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI7tDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eItsD2B,QJssDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIztDuB,QJytDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI/tDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIxtD2B,QJwtDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI3tDuB,QJ2tDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIjuDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI5tD2B,QJ4tDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI/tDuB,QJ+tDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIruDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIhuD2B,QJguDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QInuDuB,QJmuDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIzuDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIluD2B,QJkuDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIruDuB,QJquDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI3uDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIpuD2B,QJouDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIvuDuB,QJuuDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI7uDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eItuD2B,QJsuDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIzuDuB,QJyuDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MI/uDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIxuD2B,QJwuDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI3uDuB,QJ2uDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIjvDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eI5uD2B,QJ4uDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QI/uDuB,QJ+uDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIrvDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIhvD2B,QJgvDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QInvDuB,QJmvDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QAClB,qBAAO,O;;;MIzvDP,yB;mFAGJ,+B;MAQmB,kBAAR,iB;MAAQ,sB;;QJkoDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eIlvD2B,QJkvDZ,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIrvDuB,QJqvDf,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;

lB,qBAAO,O;;;MIryDP,yB;K;mFAGJ,yB;MJqyDA,sE;MFl4DA,iB;MM6FA,sC;QJkzDI,eIryDO,iBJqyDQ,W;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,MAAM,6B;QAC/B,eIvyDqB,QJuyDN,CAAS,QAAS,OAAlB,C;QACf,OAAO,QAAS,UAAhB,C;UACI,QIzyDiB,QJyyDT,CAAS,QAAS,OAAlB,C;UACR,WF54DG,MAAO,KE44DO,QF54DP,EE44DiB,CF54DjB,C;;QMkGd,OJ4yDO,Q;O;KIzzDX,C;mFAgBA,yB;MJ4yDA,sE;MFp6DA,iB;MMwHA,sC;QJyzDI,eI5yDO,iBJ4yDQ,W;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,MAAM,6B;QAC/B,eI9yDqB,QJ8yDN,CAAS,QAAS,OAAlB,C;QACf,OAAO,QAAS,UAAhB,C;UACI,QIhzDiB,QJgzDT,CAAS,QAAS,OAAlB,C;UACR,WF96DG,MAAO,KE86DO,QF96DP,EE86DiB,CF96DjB,C;;QM6Hd,OJmzDO,Q;O;KIh0DX,C;mFAgBA,yB;MJmzDA,sE;MInzDA,sC;QJ8zDI,eInzDO,iBJmzDQ,W;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,MAAM,6B;QAC/B,eIrzDqB,QJqzDN,CAAS,QAAS,OAAlB,C;QACf,OAAO,QAAS,UAAhB,C;UACI,QIvzDiB,QJuzDT,CAAS,QAAS,OAAlB,C;UACR,IAAI,2BAAW,CAAX,C;YACI,WAAW,C;;QIzzDnB,OJ4zDO,Q;O;KIv0DX,C;+FAcA,yB;MN3IA,iB;MM2IA,sC;QAWmB,kBAAR,iB;QAAQ,sB;;UJ4zDf,eAAe,sB;UACf,IAAI,CAAC,QAAS,UAAd,C;YAAyB,qBAAO,I;YAAP,uB;;UACzB,eI9zD2B,QJ8zDZ,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIh0DuB,QJg0Df,CAAS,QAAS,OAAlB,C;YACR,WF/8DG,MAAO,KE+8DO,QF/8DP,EE+8DiB,CF/8DjB,C;;UMiJd,eIn0D2B,QJm0DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIr0DuB,QJq0Df,CAAS,QAAS,OAAlB,C;YACR,WFp9DG,MAAO,KEo9DO,QFp9DP,EEo9DiB,CFp9DjB,C;;UMsKd,eIx0D2B,QJw0DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI10DuB,QJ00Df,CAAS,QAAS,OAAlB,C;YACR,WFz+DG,MAAO,KEy+DO,QFz+DP,EEy+DiB,CFz+DjB,C;;UM2Ld,eI70D2B,QJ60DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI/0DuB,QJ+0Df,CAAS,QAAS,OAAlB,C;YACR,WF9/DG,MAAO,KE8/DO,QF9/DP,EE8/DiB,CF9/DjB,C;;UMgMd,eIl1D2B,QJk1DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIp1DuB,QJo1Df,CAAS,QAAS,OAAlB,C;YACR,WFngEG,MAAO,KEmgEO,QFngEP,EEmgEiB,CFngEjB,C;;UMqMd,eIv1D2B,QJu1DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIz1DuB,QJy1Df,CAAS,QAAS,OAAlB,C;YACR,WFxhEG,MAAO,KEwhEO,QFxhEP,EEwhEiB,CFxhEjB,C;;UM0Md,eI51D2B,QJ41DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI91DuB,QJ81Df,CAAS,QAAS,OAAlB,C;YACR,WF7iEG,MAAO,KE6iEO,QF7iEP,EE6iEiB,CF7iEjB,C;;UM+Md,eIj2D2B,QJi2DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIn2DuB,QJm2Df,CAAS,QAAS,OAAlB,C;YACR,WFljEG,MAAO,KEkjEO,QFljEP,EEkjEiB,CFljEjB,C;;UMoNd,eIt2D2B,QJs2DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIx2DuB,QJw2Df,CAAS,QAAS,OAAlB,C;YACR,WFvkEG,MAAO,KEukEO,QFvkEP,EEukEiB,CFvkEjB,C;;UMyNd,eI32D2B,QJ22DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI72DuB,QJ62Df,CAAS,QAAS,OAAlB,C;YACR,WF5lEG,MAAO,KE4lEO,QF5lEP,EE4lEiB,CF5lEjB,C;;UM8Nd,eIh3D2B,QJg3DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIl3DuB,QJk3Df,CAAS,QAAS,OAAlB,C;YACR,WFjnEG,MAAO,KEinEO,QFjnEP,EEinEiB,CFjnEjB,C;;UMmOd,eIr3D2B,QJq3DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIv3DuB,QJu3Df,CAAS,QAAS,OAAlB,C;YACR,WFtoEG,MAAO,KEsoEO,QFtoEP,EEsoEiB,CFtoEjB,C;;UMwOd,eI13D2B,QJ03DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI53DuB,QJ43Df,CAAS,QAAS,OAAlB,C;YACR,WF3pEG,MAAO,KE2pEO,QF3pEP,EE2pEiB,CF3pEjB,C;;UM6Pd,eI/3D2B,QJ+3DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIj4DuB,QJi4Df,CAAS,QAAS,OAAlB,C;YACR,WFhrEG,MAAO,KEgrEO,QFhrEP,EEgrEiB,CFhrEjB,C;;UMkQd,eIp4D2B,QJo4DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIt4DuB,QJs4Df,CAAS,QAAS,OAAlB,C;YACR,WFrsEG,MAAO,KEqsEO,QFrsEP,EEqsEiB,CFrsEjB,C;;UMuQd,eIz4D2B,QJy4DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI34DuB,QJ24Df,CAAS,QAAS,OAAlB,C;YACR,WF1tEG,MAAO,KE0tEO,QF1tEP,EE0tEiB,CF1tEjB,C;;UM4Qd,eI94D2B,QJ84DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIh5DuB,QJg5Df,CAAS,QAAS,OAAlB,C;YACR,WF/uEG,MAAO,KE+uEO,QF/uEP,EE+uEiB,CF/uEjB,C;;UMiRd,eIn5D2B,QJm5DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIr5DuB,QJq5Df,CAAS,QAAS,OAAlB,C;YACR,WFpwEG,MAAO,KEowEO,QFpwEP,EEowEiB,CFpwEjB,C;;UMsRd,eIx5D2B,QJw5DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI15DuB,QJ05Df,CAAS,QAAS,OAAlB,C;YACR,WFzxEG,MAAO,KEyxEO,QFzxEP,EEyxEiB,CFzxEjB,C;;UM2Rd,eI75D2B,QJ65DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI/5DuB,QJ+5Df,CAAS,QAAS,OAAlB,C;YACR,WF9yEG,MAAO,KE8yEO,QF9yEP,EE8yEiB,CF9yEjB,C;;UMgSd,eIl6D2B,QJk6DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIp6DuB,QJo6Df,CAAS,QAAS,OAAlB,C;YACR,WFn0EG,MAAO,KEm0EO,QFn0EP,EEm0EiB,CFn0EjB,C;;UMqSd,eIv6D2B,QJu6DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIz6DuB,QJy6Df,CAAS,QAAS,OAAlB,C;YACR,WFx1EG,MAAO,KEw1EO,QFx1EP,EEw1EiB,CFx1EjB,C;;UM0Sd,eI56D2B,QJ46DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI96DuB,QJ86Df,CAAS,QAAS,OAAlB,C;YACR,WF72EG,MAAO,KE62EO,QF72EP,EE62EiB,CF72EjB,C;;UM+Sd,eIj7D2B,QJi7DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIn7DuB,QJm7Df,CAAS,QAAS,OAAlB,C;YACR,WFl4EG,MAAO,KEk4EO,QFl4EP,EEk4EiB,CFl4EjB,C;;UMoTd,eIt7D2B,QJs7DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIx7DuB,QJw7Df,CAAS,QAAS,OAAlB,C;YACR,WFv5EG,MAAO,KEu5EO,QFv5EP,EEu5EiB,CFv5EjB,C;;UMyTd,eI37D2B,QJ27DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI77DuB,QJ67Df,CAAS,QAAS,OAAlB,C;YACR,WF56EG,MAAO,KE46EO,QF56EP,EE46EiB,CF56EjB,C;;UM8Td,eIh8D2B,QJg8DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIl8DuB,QJk8Df,CAAS,QAAS,OAAlB,C;YACR,WFj8EG,MAAO,KEi8EO,QFj8EP,EEi8EiB,CFj8EjB,C;;UMmUd,eIr8D2B,QJq8DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIv8DuB,QJu8Df,CAAS,QAAS,OAAlB,C;YACR,WFt9EG,MAAO,KEs9EO,QFt9EP,EEs9EiB,CFt9EjB,C;;UMwUd,eI18D2B,QJ08DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI58DuB,QJ48Df,CAAS,QAAS,OAAlB,C;YACR,WF3+EG,MAAO,KE2+EO,QF3+EP,EE2+EiB,CF3+EjB,C;;UM6Vd,eI/8D2B,QJ+8DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIj9DuB,QJi9Df,CAAS,QAAS,OAAlB,C;YACR,WFhgFG,MAAO,KEggFO,QFhgFP,EEggFiB,CFhgFjB,C;;UMkWd,eIp9D2B,QJo9DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIt9DuB,QJs9Df,CAAS,QAAS,OAAlB,C;YACR,WFrhFG,MAAO,KEqhFO,QFrhFP,EEqhFiB,CFrhFjB,C;;UMuWd,eIz9D2B,QJy9DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI39DuB,QJ29Df,CAAS,QAAS,OAAlB,C;YACR,WF1iFG,MAAO,KE0iFO,QF1iFP,EE0iFiB,CF1iFjB,C;;UM4Wd,eI99D2B,QJ89DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIh+DuB,QJg+Df,CAAS,QAAS,OAAlB,C;YACR,WF/jFG,MAAO,KE+jFO,QF/jFP,EE+jFiB,CF/jFjB,C;;UMiXd,eIn+D2B,QJm+DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIr+DuB,QJq+Df,CAAS,QAAS,OAAlB,C;YACR,WFplFG,MAAO,KEolFO,QFplFP,EEolFiB,CFplFjB,C;;UMsXd,eIx+D2B,QJw+DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI1+DuB,QJ0+Df,CAAS,QAAS,OAAlB,C;YACR,WFzmFG,MAAO,KEymFO,QFzmFP,EEymFiB,CFzmFjB,C;;UM2Xd,eI7+D2B,QJ6+DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI/+DuB,QJ++Df,CAAS,QAAS,OAAlB,C;YACR,WF9nFG,MAAO,KE8nFO,QF9nFP,EE8nFiB,CF9nFjB,C;;UMgYd,eIl/D2B,QJk/DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIp/DuB,QJo/Df,CAAS,QAAS,OAAlB,C;YACR,WFnpFG,MAAO,KEmpFO,QFnpFP,EEmpFiB,CFnpFjB,C;;UMqYd,eIv/D2B,QJu/DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIz/DuB,QJy/Df,CAAS,QAAS,OAAlB,C;YACR,WFxqFG,MAAO,KEwqFO,QFxqFP,EEwqFiB,CFxqFjB,C;;UM0Yd,eI5/D2B,QJ4/DpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI9/DuB,QJ8/Df,CAAS,QAAS,OAAlB,C;YACR,WF7rFG,MAAO,KE6rFO,QF7rFP,EE6rFiB,CF7rFjB,C;;UM+Yd,eIjgE2B,QJigEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIngEuB,QJmgEf,CAAS,QAAS,OAAlB,C;YACR,WFlsFG,MAAO,KEksFO,QFlsFP,EEksFiB,CFlsFjB,C;;UMoZd,eItgE2B,QJsgEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIxgEuB,QJwgEf,CAAS,QAAS,OAAlB,C;YACR,WFvtFG,MAAO,KEutFO,QFvtFP,EEutFiB,CFvtFjB,C;;UMyZd,eI3gE2B,QJ2gEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI7gEuB,QJ6gEf,CAAS,QAAS,OAAlB,C;YACR,WF5uFG,MAAO,KE4uFO,QF5uFP,EE4uFiB,CF5uFjB,C;;UM8Zd,eIhhE2B,QJghEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIlhEuB,QJkhEf,CAAS,QAAS,OAAlB,C;YACR,WFjvFG,MAAO,KEivFO,QFjvFP,EEivFiB,CFjvFjB,C;;UMmad,eIrhE2B,QJqhEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIvhEuB,QJuhEf,CAAS,QAAS,OAAlB,C;YACR,WFtwFG,MAAO,KEswFO,QFtwFP,EEswFiB,CFtwFjB,C;;UMwad,eI1hE2B,QJ0hEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI5hEuB,QJ4hEf,CAAS,QAAS,OAAlB,C;YACR,WF3xFG,MAAO,KE2xFO,QF3xFP,EE2xFiB,CF3xFjB,C;;UM6ad,eI/hE2B,QJ+hEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIjiEuB,QJiiEf,CAAS,QAAS,OAAlB,C;YACR,WFhzFG,MAAO,KEgzFO,QFhzFP,EEgzFiB,CFhzFjB,C;;UMkbd,eIpiE2B,QJoiEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QItiEuB,QJsiEf,CAAS,QAAS,OAAlB,C;YACR,WFr0FG,MAAO,KEq0FO,QFr0FP,EEq0FiB,CFr0FjB,C;;UMubd,eIziE2B,QJyiEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI3iEuB,QJ2iEf,CAAS,QAAS,OAAlB,C;YACR,WF11FG,MAAO,KE01FO,QF11FP,EE01FiB,CF11FjB,C;;UM4bd,eI9iE2B,QJ8iEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIhjEuB,QJgjEf,CAAS,QAAS,OAAlB,C;YACR,WF/2FG,MAAO,KE+2FO,QF/2FP,EE+2FiB,CF/2FjB,C;;UMicd,eInjE2B,QJmjEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIrjEuB,QJqjEf,CAAS,QAAS,OAAlB,C;YACR,WFp4FG,MAAO,KEo4FO,QFp4FP,EEo4FiB,CFp4FjB,C;;UMsdd,eIxjE2B,QJwjEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI1jEuB,QJ0jEf,CAAS,QAAS,OAAlB,C;YACR,WFz5FG,MAAO,KEy5FO,QFz5FP,EEy5FiB,CFz5FjB,C;;UM2dd,eI7jE2B,QJ6jEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI/jEuB,QJ+jEf,CAAS,QAAS,OAAlB,C;YACR,WF96FG,MAAO,KE86FO,QF96FP,EE86FiB,CF96FjB,C;;UMged,eIlkE2B,QJkkEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIpkEuB,QJokEf,CAAS,QAAS,OAAlB,C;YACR,WFn8FG,MAAO,KEm8FO,QFn8FP,EEm8FiB,CFn8FjB,C;;UMqed,eIvkE2B,QJukEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIzkEuB,QJykEf,CAAS,QAAS,OAAlB,C;YACR,WFx9FG,MAAO,KEw9FO,QFx9FP,EEw9FiB,CFx9FjB,C;;UM0ed,eI5kE2B,QJ4kEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI9kEuB,QJ8kEf,CAAS,QAAS,OAAlB,C;YACR,WF7+FG,MAAO,KE6+FO,QF7+FP,EE6+FiB,CF7+FjB,C;;UM+fd,eIjlE2B,QJilEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QInlEuB,QJmlEf,CAAS,QAAS,OAAlB,C;YACR,WFlgGG,MAAO,KEkgGO,QFlgGP,EEkgGiB,CFlgGjB,C;;UMohBd,eItlE2B,QJslEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIxlEuB,QJwlEf,CAAS,QAAS,OAAlB,C;YACR,WFvhGG,MAAO,KEuhGO,QFvhGP,EEuhGiB,CFvhGjB,C;;UMyhBd,eI3lE2B,QJ2lEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI7lEuB,QJ6lEf,CAAS,QAAS,OAAlB,C;YACR,WF5iGG,MAAO,KE4iGO,QF5iGP,EE4iGiB,CF5iGjB,C;;UM8iBd,eIhmE2B,QJgmEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIlmEuB,QJkmEf,CAAS,QAAS,OAAlB,C;YACR,WFjkGG,MAAO,KEikGO,QFjkGP,EEikGiB,CFjkGjB,C;;UMmjBd,eIrmE2B,QJqmEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIvmEuB,QJumEf,CAAS,QAAS,OAAlB,C;YACR,WFtlGG,MAAO,KEslGO,QFtlGP,EEslGiB,CFtlGjB,C;;UMwjBd,eI1mE2B,QJ0mEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI5mEuB,QJ4mEf,CAAS,QAAS,OAAlB,C;YACR,WF3mGG,MAAO,KE2mGO,QF3mGP,EE2mGiB,CF3mGjB,C;;UM6jBd,eI/mE2B,QJ+mEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIjnEuB,QJinEf,CAAS,QAAS,OAAlB,C;YACR,WFhoGG,MAAO,KEgoGO,QFhoGP,EEgoGiB,CFhoGjB,C;;UMkkBd,eIpnE2B,QJonEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QItnEuB,QJsnEf,CAAS,QAAS,OAAlB,C;YACR,WFrpGG,MAAO,KEqpGO,QFrpGP,EEqpGiB,CFrpGjB,C;;UMulBd,eIznE2B,QJynEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI3nEuB,QJ2nEf,CAAS,QAAS,OAAlB,C;YACR,WF1qGG,MAAO,KE0qGO,QF1qGP,EE0qGiB,CF1qGjB,C;;UM4lBd,eI9nE2B,QJ8nEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIhoEuB,QJgoEf,CAAS,QAAS,OAAlB,C;YACR,WF/rGG,MAAO,KE+rGO,QF/rGP,EE+rGiB,CF/rGjB,C;;UMinBd,eInoE2B,QJmoEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIroEuB,QJqoEf,CAAS,QAAS,OAAlB,C;YACR,WFptGG,MAAO,KEotGO,QFptGP,EEotGiB,CFptGjB,C;;UMsnBd,eIxoE2B,QJwoEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI1oEuB,QJ0oEf,CAAS,QAAS,OAAlB,C;YACR,WFzuGG,MAAO,KEyuGO,QFzuGP,EEyuGiB,CFzuGjB,C;;UM2nBd,eI7oE2B,QJ6oEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI/oEuB,QJ+oEf,CAAS,QAAS,OAAlB,C;YACR,WF9vGG,MAAO,KE8vGO,QF9vGP,EE8vGiB,CF9vGjB,C;;UMgoBd,eIlpE2B,QJkpEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIppEuB,QJopEf,CAAS,QAAS,OAAlB,C;YACR,WFnxGG,MAAO,KEmxGO,QFnxGP,EEmxGiB,CFnxGjB,C;;UMqoBd,eIvpE2B,QJupEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIzpEuB,QJypEf,CAAS,QAAS,OAAlB,C;YACR,WFxyGG,MAAO,KEwyGO,QFxyGP,EEwyGiB,CFxyGjB,C;;UM0oBd,eI5pE2B,QJ4pEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI9pEuB,QJ8pEf,CAAS,QAAS,OAAlB,C;YACR,WF7zGG,MAAO,KE6zGO,QF7zGP,EE6zGiB,CF7zGjB,C;;UM+pBd,eIjqE2B,QJiqEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QInqEuB,QJmqEf,CAAS,QAAS,OAAlB,C;YACR,WFl1GG,MAAO,KEk1GO,QFl1GP,EEk1GiB,CFl1GjB,C;;UMoqBd,eItqE2B,QJsqEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIxqEuB,QJwqEf,CAAS,QAAS,OAAlB,C;YACR,WFv2GG,MAAO,KEu2GO,QFv2GP,EEu2GiB,CFv2GjB,C;;UMyqBd,eI3qE2B,QJ2qEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI7qEuB,QJ6qEf,CAAS,QAAS,OAAlB,C;YACR,WF53GG,MAAO,KE43GO,QF53GP,EE43GiB,CF53GjB,C;;UM8rBd,eIhrE2B,QJgrEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIlrEuB,QJkrEf,CAAS,QAAS,OAAlB,C;YACR,WFj4GG,MAAO,KEi4GO,QFj4GP,EEi4GiB,CFj4GjB,C;;UMmsBd,eIrrE2B,QJqrEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIvrEuB,QJurEf,CAAS,QAAS,OAAlB,C;YACR,WFt5GG,MAAO,KEs5GO,QFt5GP,EEs5GiB,CFt5GjB,C;;UMwsBd,eI1rE2B,QJ0rEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI5rEuB,QJ4rEf,CAAS,QAAS,OAAlB,C;YACR,WF36GG,MAAO,KE26GO,QF36GP,EE26GiB,CF36GjB,C;;UM6sBd,eI/rE2B,QJ+rEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIjsEuB,QJisEf,CAAS,QAAS,OAAlB,C;YACR,WFh8GG,MAAO,KEg8GO,QFh8GP,EEg8GiB,CFh8GjB,C;;UMktBd,eIpsE2B,QJosEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QItsEuB,QJssEf,CAAS,QAAS,OAAlB,C;YACR,WFr9GG,MAAO,KEq9GO,QFr9GP,EEq9GiB,CFr9GjB,C;;UMutBd,eIzsE2B,QJysEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI3sEuB,QJ2sEf,CAAS,QAAS,OAAlB,C;YACR,WF1+GG,MAAO,KE0+GO,QF1+GP,EE0+GiB,CF1+GjB,C;;UM4tBd,eI9sE2B,QJ8sEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIhtEuB,QJgtEf,CAAS,QAAS,OAAlB,C;YACR,WF/gHG,MAAO,KE+gHO,QF/gHP,EE+gHiB,CF/gHjB,C;;UMiuBd,eIntE2B,QJmtEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIrtEuB,QJqtEf,CAAS,QAAS,OAAlB,C;YACR,WFphHG,MAAO,KEohHO,QFphHP,EEohHiB,CFphHjB,C;;UMsuBd,eIxtE2B,QJwtEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI1tEuB,QJ0tEf,CAAS,QAAS,OAAlB,C;YACR,WFziHG,MAAO,KEyiHO,QFziHP,EEyiHiB,CFziHjB,C;;UM2uBd,eI7tE2B,QJ6tEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI/tEuB,QJ+tEf,CAAS,QAAS,OAAlB,C;YACR,WF9jHG,MAAO,KE8jHO,QF9jHP,EE8jHiB,CF9jHjB,C;;UMgvBd,eIluE2B,QJkuEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIpuEuB,QJouEf,CAAS,QAAS,OAAlB,C;YACR,WFnlHG,MAAO,KEmlHO,QFnlHP,EEmlHiB,CFnlHjB,C;;UMqvBd,eIvuE2B,QJuuEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIzuEuB,QJyuEf,CAAS,QAAS,OAAlB,C;YACR,WFxmHG,MAAO,KEwmHO,QFxmHP,EEwmHiB,CFxmHjB,C;;UM0vBd,eI5uE2B,QJ4uEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI9uEuB,QJ8uEf,CAAS,QAAS,OAAlB,C;YACR,WF7nHG,MAAO,KE6nHO,QF7nHP,EE6nHiB,CF7nHjB,C;;UM+vBd,eIjvE2B,QJivEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QInvEuB,QJmvEf,CAAS,QAAS,OAAlB,C;YACR,WFlpHG,MAAO,KEkpHO,QFlpHP,EEkpHiB,CFlpHjB,C;;UMowBd,eItvE2B,QJsvEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIxvEuB,QJwvEf,CAAS,QAAS,OAAlB,C;YACR,WFvqHG,MAAO,KEuqHO,QFvqHP,EEuqHiB,CFvqHjB,C;;UMyxBd,eI3vE2B,QJ2vEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI7vEuB,QJ6vEf,CAAS,QAAS,OAAlB,C;YACR,WF5rHG,MAAO,KE4rHO,QF5rHP,EE4rHiB,CF5rHjB,C;;UM8xBd,eIhwE2B,QJgwEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIlwEuB,QJkwEf,CAAS,QAAS,OAAlB,C;YACR,WFjsHG,MAAO,KEisHO,QFjsHP,EEisHiB,CFjsHjB,C;;UMmyBd,eIrwE2B,QJqwEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIvwEuB,QJuwEf,CAAS,QAAS,OAAlB,C;YACR,WFttHG,MAAO,KEstHO,QFttHP,EEstHiB,CFttHjB,C;;UMwyBd,eI1wE2B,QJ0wEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI5wEuB,QJ4wEf,CAAS,QAAS,OAAlB,C;YACR,WF3uHG,MAAO,KE2uHO,QF3uHP,EE2uHiB,CF3uHjB,C;;UM6yBd,eI/wE2B,QJ+wEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIjxEuB,QJixEf,CAAS,QAAS,OAAlB,C;YACR,WFhvHG,MAAO,KEgvHO,QFhvHP,EEgvHiB,CFhvHjB,C;;UMkzBd,eIpxE2B,QJoxEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QItxEuB,QJsxEf,CAAS,QAAS,OAAlB,C;YACR,WFrwHG,MAAO,KEqwHO,QFrwHP,EEqwHiB,CFrwHjB,C;;UMuzBd,eIzxE2B,QJyxEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI3xEuB,QJ2xEf,CAAS,QAAS,OAAlB,C;YACR,WF1xHG,MAAO,KE0xHO,QF1xHP,EE0xHiB,CF1xHjB,C;;UM4zBd,eI9xE2B,QJ8xEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIhyEuB,QJgyEf,CAAS,QAAS,OAAlB,C;YACR,WF/yHG,MAAO,KE+yHO,QF/yHP,EE+yHiB,CF/yHjB,C;;UMi0Bd,eInyE2B,QJmyEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIryEuB,QJqyEf,CAAS,QAAS,OAAlB,C;YACR,WFp0HG,MAAO,KEo0HO,QFp0HP,EEo0HiB,CFp0HjB,C;;UMs0Bd,eIxyE2B,QJwyEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI1yEuB,QJ0yEf,CAAS,QAAS,OAAlB,C;YACR,WFz1HG,MAAO,KEy1HO,QFz1HP,EEy1HiB,CFz1HjB,C;;UM20Bd,eI7yE2B,QJ6yEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QI/yEuB,QJ+yEf,CAAS,QAAS,OAAlB,C;YACR,WF92HG,MAAO,KE82HO,QF92HP,EE82HiB,CF92HjB,C;;UMg1Bd,eIlzE2B,QJkzEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIpzEuB,QJozEf,CAAS,QAAS,OAAlB,C;YACR,WFn3HG,MAAO,KEm3HO,QFn3HP,EEm3HiB,CFn3HjB,C;;UMq1Bd,eIvzE2B,QJuzEpB,CAAS,QAAS,OAAlB,C;UACf,OAAO,QAAS,UAAhB,C;YACI,QIzzEuB,QJyzEf,CAAS,QAAS,OAAlB,C;YACR,WFx4HG,MAAO,KEw4HO,QFx4HP,EEw4HiB,CFx4

OI,OAAO,wBAAa,cAAb,C;O;KAPX,C;4FAUA,yB;MAAA,mC;MAAA,uD;MAAA,4B;QAOI,OAAO,wBAAa,cAAb,C;O;KAPX,C;IAUA,4C;MAMI,IAAI,mBAAJ,C;QACI,OAAO,I;MACX,OAAc,QAAP,MAAO,EAAQ,SAAR,C;K;IAGlB,4C;MAMI,IAAI,mBAAJ,C;QACI,OAAO,I;MACX,OAAc,SAAP,MAAO,EAAS,SAAT,C;K;IAGlB,4C;MAMI,IAAI,mBAAJ,C;QACI,OAAO,I;MACX,OAAiD,OAA1C,MAAO,iBAAQ,e4BxGoB,I5BwG5B,5B,EAAoB,CAAA,c4BxGQ,I5BwGR,GAAY,CAAZ,EAAe,IAAf,pB,CAAmC,C;K;mFAGrD,8B;MAQI,OAAO,mBAAmB,2BAAS,SAAT,C;K;oFAG9B,8B;MAQI,OAAO,mBAAmB,2BAAS,SAAT,C;K;oFAG9B,8B;MAQI,OAAO,mBAAmB,2BAAS,SAAT,C;K;IAG9B,uC;MAKI,OAAO,2BAAe,KAAf,C;K;IAGX,uC;MAKI,OAAO,2BAAe,oBAAf,EAAqB,KAArB,EAA2B,CAA3B,C;K;IAGX,uC;MAKI,OAAO,2BAAe,KAAf,C;K;IAGX,uC;MAOI,OAAO,2BAAe,KAAf,C;K;IAGX,uC;MAOI,OAAO,2BAAe,KAAf,C;K;IAGX,uC;MgBzHW,ShBgIM,mBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MgBnIW,ShB0IM,kBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MgB7IW,ShBoJM,oBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MgBvJW,ShB8JM,qBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MAKI,OAAO,2BAAe,KAAf,C;K;IAGX,uC;MgBzKW,ShBgLM,mBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MgBnLW,ShB0LM,kBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MgB7LW,ShBoMM,oBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MgBvMW,ShB8MM,qBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MAKI,OAAO,2BAAe,KAAf,C;K;IAGX,uC;MgBzNW,ShBgOM,mBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MgBnOW,ShB0OM,kBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MgB7OW,ShBoPM,oBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MgBvPW,ShB8PM,qBAAN,EAAa,KAAb,EAAmB,C;MAAnB,OAAuB,CAAvB,C0B0Ca,O1BzCb,C;K;IAGX,uC;MAKI,OAAO,2BAAe,KAAf,C;K;IAGX,uC;MAO

AA8B,IAA9B,C;MACA,OAAO,uCAAgB,yBAAgB,eAAhB,EAAuB,cAAvB,EAAiC,SAAK,KAAL,GAAY,CAAh
B,GAAmB,IAAnB,GAA6B,CAAC,IAAD,IAA1D,C;K;IAG3B,sC;MACI,OAAmB,IAAR,8BAAgC,GAApC,GAAi
E,OAAL,SAAK,CAAjE,GAA+E,I;K;IAG1F,wC;MACI,OAAW,mEAAJ,GAAmE,OAAL,SAAK,SAAnE,GAAiF,I;
K;IAG5F,wC;MACI,OAAW,YAAQ,aAAA,sCAAe,UAAf,EAA0B,sCAAe,UAAzC,CAAR,YAAJ,GAAqE,OAAL,
SAAK,CAArE,GAAmF,I;K;IAG9F,wC;MACI,OAAmB,UAAA,sCAAe,UAAf,EAA2B,sCAAe,UAA1C,CAAR,4B
AAJ,GAA+E,OAAR,YAAL,SAAK,CAAQ,CAA/E,GAA6F,I;K;IAGxG,wC;MACI,OAAmB,UAAA,sCAAe,UAAf
,EAA0B,sCAAe,UAAzC,CAAR,4BAAJ,GAA6E,OAAR,YAAL,SAAK,CAAQ,CAA7E,GAA2F,I;K;IAGtG,qC;M
ACI,OAAW,iFAAJ,GAA4D,SAAK,QAAjE,GAA8E,I;K;IAGzF,uC;MACI,OAAmB,UAAc,WAAd,EAAwC,UAA
xC,CAAR,4BAAJ,GAAqE,YAAL,SAAK,CAArE,GAAkF,I;K;IAG7F,uC;MACI,OAAmB,UAAc,WAAd,EAAuC,
UAAvC,CAAR,4BAAJ,GAAmE,YAAL,SAAK,CAAnE,GAAgF,I;K;IAG3F,sC;MACI,OAAmB,UAAe,mCAAf,E
AA0C,mCAA1C,CAAR,4BAAJ,GAAuE,uBAAL,SAAK,CAAvE,GAAqF,I;K;IAGhG,wC;MACI,OAAmB,UAAe,
mCAAf,EAAyC,mCAAzC,CAAR,4BAAJ,GAAqE,uBAAL,SAAK,CAArE,GAAmF,I;K;IAG9F,uC;MACI,OAAm
B,MAAR,8BAAiC,KAArC,GAAmE,QAAL,SAAK,CAAnE,GAAkF,I;K;IAG7F,yC;MACI,OAAW,uEAAJ,GAAq
E,QAAL,SAAK,SAArE,GAAoF,I;K;IAG/F,yC;MACI,OAAmB,UAAA,uCAAgB,UAAhB,EAA4B,uCAAgB,UAA
5C,CAAR,4BAAJ,GAAiF,QAAR,YAAL,SAAK,CAAQ,CAAjF,GAAgG,I;K;IAG3G,yC;MACI,OAAmB,UAAA,u
CAAgB,UAAhB,EAA2B,uCAAgB,UAA3C,CAAR,4BAAJ,GAA+E,QAAR,YAAL,SAAK,CAAQ,CAA/E,GAA8F
,I;K;IAGzG,8B;MAMI,OAAO,wBAAY,EAAa,GAAH,CAAG,IAAzB,C;K;IAGX,gC;MAMI,OAAO,kBAAY,oBA
AH,EAAG,CAAc,8BAAH,CAAG,EAA1B,C;K;IAGX,gC;MAMI,OAAO,aAAK,SAAL,EAAoB,EAAa,GAAH,CA
AG,IAAjC,C;K;IAGX,gC;MAMI,OAAO,aAAK,SAAL,EAAoB,EAAa,GAAH,CAAG,IAAjC,C;K;IAGX,gC;MA
MI,IAAI,MAAM,CAAV,C;QAAoB,OAAO,iCAAU,M;MACrC,OAAO,yBAAiB,OAAR,EAAQ,GAAH,CAAG,CA
AjB,C;K;IAGX,gC;MAMI,IAAI,MAAM,WAAV,C;QAAyB,OAAO,gCAAS,M;MACzC,OAAO,wBAAS,EAAQ,G
AAH,CAAG,IAAjB,C;K;IAGX,gC;MAMI,OAAO,kBAAY,oBAAH,EAAG,CAAc,8BAAH,CAAG,EAA1B,C;K;I
AGX,gC;MAMI,IAAI,MAAM,WAAV,C;QAAyB,OAAO,gCAAS,M;MACzC,OAAO,aAAK,SAAL,EAAiB,EAA
Q,GAAH,CAAG,IAAzB,C;K;IAGX,gC;MAMI,IAAI,MAAM,WAAV,C;QAAyB,OAAO,gCAAS,M;MACzC,OA
AO,aAAK,SAAL,EAAiB,EAAQ,GAAH,CAAG,IAAzB,C;K;IAGX,gC;MAMI,IAAI,iDAAJ,C;QAA0B,OAAO,iC
AAU,M;MAC3C,OAAY,oBAAL,SAAK,CAAL,SAAkB,EAAQ,8BAAH,CAAG,EAA1B,C;K;IAGX,gC;MAMI,IA
AI,iDAAJ,C;QAA0B,OAAO,iCAAU,M;MAC3C,OAAO,kBAAS,EAAQ,8BAAH,CAAG,EAAjB,C;K;IAGX,iC;M
AMI,IAAI,iDAAJ,C;QAA0B,OAAO,iCAAU,M;MAC3C,OAAY,oBAAL,SAAK,CAAL,SAAkB,EAAQ,8BAAH,C
AAG,EAA1B,C;K;IAGX,iC;MAMI,IAAI,iDAAJ,C;QAA0B,OAAO,iCAAU,M;MAC3C,OAAY,oBAAL,SAAK,C
AAL,SAAkB,EAAQ,8BAAH,CAAG,EAA1B,C;K;IAGX,iC;MAMI,OAAO,wBAAY,EAAa,GAAH,CAAG,IAAzB
,C;K;IAGX,iC;MAMI,OAAO,kBAAY,oBAAH,EAAG,CAAc,8BAAH,CAAG,EAA1B,C;K;IAGX,iC;MAMI,OAA
O,aAAK,SAAL,EAAoB,EAAa,GAAH,CAAG,IAAjC,C;K;IAGX,iC;MAMI,OAAO,aAAK,SAAL,EAAoB,EAAa,
GAAH,CAAG,IAAjC,C;K;IAGX,gD;MAQI,OAAW,4BAAO,YAAP,KAAJ,GAAyB,YAAzB,GAA2C,S;K;IAGtD,
kD;MAQI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;K;IAGtD,kD;MAQI,OAAW,YAAO,YAAX,GAAy
B,YAAzB,GAA2C,S;K;IAGtD,kD;MAQI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;K;IAGtD,kD;MAQI
,OAAW,0BAAO,YAAP,KAAJ,GAAyB,YAAzB,GAA2C,S;K;IAGtD,kD;MAQI,OAAW,YAAO,YAAX,GAAyB,
YAAzB,GAA2C,S;K;IAGtD,kD;MAQI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;K;IAGtD,+C;MAQI,O
AAW,4BAAO,YAAP,KAAJ,GAAyB,YAAzB,GAA2C,S;K;IAGtD,iD;MAQI,OAAW,YAAO,YAAX,GAAyB,YA
AzB,GAA2C,S;K;IAGtD,iD;MAQI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;K;IAGtD,iD;MAQI,OAA
W,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;K;IAGtD,iD;MAQI,OAAW,0BAAO,YAAP,KAAJ,GAAyB,YAAzB
,GAA2C,S;K;IAGtD,iD;MAQI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;K;IAGtD,iD;MAQI,OAAW,Y
AAO,YAAX,GAAyB,YAAzB,GAA2C,S;K;IAGtD,yD;MAQI,IAAI,iBAAiB,IAAjB,IAAyB,iBAAiB,IAA9C,C;Q
ACI,IAAI,+BAAe,YAAf,KAAJ,C;UAAiC,MAAM,gCAAyB,6DAAiD,YAAjD,wCAAoF,YAApF,OAAzB,C;QAC
vC,IAAI,4BAAO,YAAP,KAAJ,C;UAAyB,OAAO,Y;QAChC,IAAI,4BAAO,YAAP,KAAJ,C;UAAyB,OAAO,Y;;Q
AGhC,IAAI,iBAAiB,IAAjB,IAAyB,4BAAO,YAAP,KAA7B,C;UAAkD,OAAO,Y;QACzD,IAAI,iBAAiB,IAAjB,I
AAyB,4BAAO,YAAP,KAA7B,C;UAAkD,OAAO,Y;;MAE7D,OAAO,S;K;IAGX,2D;MAQI,IAAI,eAAe,YAAnB,
C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IAAI,YAAO,YAAX,C;QAA
yB,OAAO,Y;MAChC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MAChC,OAAO,S;K;IAGX,2D;MAQI,IAAI,eAA

e,YAAnB,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MAChC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MAChC,OAAO,S;K;IAGX,2D;MAQI,IAAI,eAAe,YAAnB,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MAChC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MAChC,OAAO,S;K;IAGX,2D;MAQI,IAAI,6BAAe,YAAf,KAAJ,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,yCAAoF,YAApF,iBAAzB ,C;MACvC,IAAI,0BAAO,YAAP,KAAJ,C;QAAyB,OAAO,Y;MAChC,IAAI,0BAAO,YAAP,KAAJ,C;QAAyB,OAAO,Y;MAChC,OAAO,S;K;IAGX,2D;MAQI,IAAI,eAAe,YAAnB,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MAChC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MAChC,OAAO,S;K;IAGX,2D;MAQI,IAAI,eAAe,YAAnB,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MAChC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MAChC,OAAO,S;K;IAGX,sC;MAUW,Q;MADP,IAAI,KAAM,UAAV,C;QAAqB,MAAM,gCAAyB,4CAAyC,4BAAzB,C;MAAyB

AAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,mBAAmB,KAAnB,C;UACA,IAAI,UAAU,IAAV,CA
AJ,C;YACI,YAAY,K;UAChB,qB;;QAEJ,OAAO,S;O;KAdX,C;IAiBA,4B;MAUI,eAAe,oB;MACf,IAAI,CAAC,Q
AAS,UAAd,C;QACI,MAAM,2BAAuB,oBAAvB,C;MACV,WAAW,QAAS,O;MACpB,OAAO,QAAS,UAAhB,C;
QACI,OAAO,QAAS,O;MACpB,OAAO,I;K;+EAGX,yB;MAAA,iE;MAAA,gB;MAAA,8B;MAAA,uC;QAYoB,U
AQT,M;QAVP,WAAe,I;QACf,YAAY,K;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,O
AAV,CAAJ,C;YACI,OAAO,O;YACP,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAAY,MAAM,gCAAuB,sDAA
vB,C;QAElB,OAAO,2E;O;KApBX,C;IAuBA,4C;MAQiB,Q;MAFb,gBAAgB,E;MAChB,YAAY,C;MACC,2B;MA
Ab,OAAa,cAAb,C;QAAa,sB;QACT,mBAAmB,KAAnB,C;QACA,IAAI,gBAAW,IAAX,CAAJ,C;UACI,YAAY,K;
QAChB,qB;;MAEJ,OAAO,S;K;IAGX,kC;MAQI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QACI,OAAO,I;M
ACX,WAAW,QAAS,O;MACpB,OAAO,QAAS,UAAhB,C;QACI,OAAO,QAAS,O;MACpB,OAAO,I;K;2FAGX,g
C;MASoB,Q;MADhB,WAAe,I;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,UAAU,OAAV,C
AAJ,C;UACI,OAAO,O;;;MAGf,OAAO,I;K;IAGX,8B;MAMI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAC
I,MAAM,2BAAuB,oBAAvB,C;MACV,aAAa,QAAS,O;MACtB,IAAI,QAAS,UAAb,C;QACI,MAAM,gCAAyB,yB;Q
CAAzB,C;MACV,OAAO,M;K;mFAGX,yB;MAAA,kF;MAAA,iE;MAAA,gB;MAAA,8B;MAAA,uC;QAQoB,UAAU,M;QAXP,aAAa,iB;QACjB,YAAY,K;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,O
AAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,mDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAG
GhB,IAAI,CAAC,KAAL,C;UAAY,MAAM,gCAAuB,sDAAvB,C;QAElB,OAAO,6E;O;KAjBX,C;IAoBA,oC;MA
MI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QACI,OAAO,I;MACX,aAAa,QAAS,O;MACtB,IAAI,QAAS,U
Ab,C;QACI,OAAO,I;MACX,OAAO,M;K;+FAGX,gC;MAQoB,Q;MAFhB,aAAa,iB;MACjB,YAAY,K;MACI,2B
;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAAW,MAAM,8BAAyB,mDAAzB,C;QAC1B,SAAS,O;QACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAAY,MAAM,gCAAuB,sDAAvB,C;MAClB,OAAO,I;K;IAGX,kC;MAQI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QACI,OAAO,I;MACX,aAAa,QAAS,O;MACtB,IAAI,QAAS,UAAb,C;QACI,MAAM,gCAAyB,yB;QAC/

IAWA,uC;MAQI,OAAO,wBAAW,cAAX,C;K;IAWA,uE;MAAA,sC;MAAA,4C;K;kDACH,Y;MACI,iBAAiC,iBAAhB,oBAAgB,C;MACtB,WAAX,UAAW,EAAS,uBAAT,C;MACX,OAAO,UAAW,W;K;;IAZ9B,6C;MAQI,0D;K;wFASJ,yB;MAAA,wE;MAAA,uC;QAaW,kBAAY,oB;QAiFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAlFsC,SAkFvB,CAAU,OAAV,C;UvBnEnB,wBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;;QuBfA,OAoFO,W;O;KAjGX,C;6FAgBA,yB;MAAA,wE;MAAA,yC;QAaW,kBAAc,oB;QA8BL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aA/B4B,WA+BxB,CAAY,OAAZ,CAAa,EAA0B,OAA1B,C;;QA/BzB,OAiCO,W;O;KA9CX,C;6FAgBA,yB;MAAA,wE;MAAA,yD;QAaW,kBAAc,oB;QAiCL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAlC4B,WAkCxB,CAAY,OAAZ,CAAa,EAAkB,OAAlB,C;UAlCjB,OAAf,CAA1B,C;;QAkCA,OAAO,W;O;KA/CX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmCL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnCjB,OAqCO,W;O;KAlDX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuCL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvCjB,OAyCO,W;O;KAtDX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2CL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3CjB,OA6CO,W;O;KA1DX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+CL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/CjB,OAiDO,W;O;KA9DX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmDL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnDjB,OAqDO,W;O;KAlEX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuDL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvDjB,OAyDO,W;O;KAtEX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2DL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3DjB,OA6DO,W;O;KA1EX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+DL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/DjB,OAiEO,W;O;KA9EX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmEL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnEjB,OAqEO,W;O;KAlFX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuEL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvEjB,OAyEO,W;O;KAtFX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2EL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3EjB,OA6EO,W;O;KA1FX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+EL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/EjB,OAiFO,W;O;KA9FX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmFL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnFjB,OAqFO,W;O;KAlGX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuFL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvFjB,OAyFO,W;O;KAtGX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2FL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3FjB,OA6FO,W;O;KA1GX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+FL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/FjB,OAiGO,W;O;KA9GX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmGL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnGjB,OAqGO,W;O;KAlHX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuGL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvGjB,OAyGO,W;O;KAtHX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2GL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3GjB,OA6GO,W;O;KA1HX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+GL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/GjB,OAiHO,W;O;KA9HX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmHL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnHjB,OAqHO,W;O;KAlIX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuHL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvHjB,OAyHO,W;O;KAtIX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2HL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3HjB,OA6HO,W;O;KA1IX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+HL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/HjB,OAiIO,W;O;KA9IX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmIL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnIjB,OAqIO,W;O;KAlJX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuIL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvIjB,OAyIO,W;O;KAtJX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2IL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3IjB,OA6IO,W;O;KA1JX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+IL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/IjB,OAiJO,W;O;KA9JX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmJL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnJjB,OAqJO,W;O;KAlKX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuJL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvJjB,OAyJO,W;O;KAtKX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2JL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3JjB,OA6JO,W;O;KA1KX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+JL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/JjB,OAiKO,W;O;KA9KX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmKL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnKjB,OAqKO,W;O;KAlLX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuKL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvKjB,OAyKO,W;O;KAtLX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2KL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3KjB,OA6KO,W;O;KA1LX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+KL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/KjB,OAiLO,W;O;KA9LX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmLL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnLjB,OAqLO,W;O;KAlMX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuLL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvLjB,OAyLO,W;O;KAtMX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2LL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3LjB,OA6LO,W;O;KA1MX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+LL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/LjB,OAiMO,W;O;KA9MX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmML,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnMjB,OAqMO,W;O;KAlNX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuML,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvMjB,OAyMO,W;O;KAtNX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2ML,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3MjB,OA6MO,W;O;KA1NX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+ML,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/MjB,OAiNO,W;O;KA9NX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmNL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnNjB,OAqNO,W;O;KAlOX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuNL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvNjB,OAyNO,W;O;KAtOX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2NL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3NjB,OA6NO,W;O;KA1OX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+NL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/NjB,OAiOO,W;O;KA9OX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmOL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnOjB,OAqOO,W;O;KAlPX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuOL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvOjB,OAyOO,W;O;KAtPX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2OL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3OjB,OA6OO,W;O;KA1PX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+OL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/OjB,OAiPO,W;O;KA9PX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmPL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnPjB,OAqPO,W;O;KAlQX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuPL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvPjB,OAyPO,W;O;KAtQX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2PL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3PjB,OA6PO,W;O;KA1QX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+PL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/PjB,OAiQO,W;O;KA9QX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmQL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnQjB,OAqQO,W;O;KAlRX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuQL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvQjB,OAyQO,W;O;KAtRX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2QL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3QjB,OA6QO,W;O;KA1RX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+QL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/QjB,OAiRO,W;O;KA9RX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmRL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnRjB,OAqRO,W;O;KAlSX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuRL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvRjB,OAyRO,W;O;KAtSX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2RL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3RjB,OA6RO,W;O;KA1SX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+RL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/RjB,OAiSO,W;O;KA9SX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmSL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnSjB,OAqSO,W;O;KAlTX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuSL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvSjB,OAySO,W;O;KAtTX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2SL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3SjB,OA6SO,W;O;KA1TX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+SL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/SjB,OAiTO,W;O;KA9TX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmTL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnTjB,OAqTO,W;O;KAlUX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuTL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvTjB,OAyTO,W;O;KAtUX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2TL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3TjB,OA6TO,W;O;KA1UX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+TL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/TjB,OAiUO,W;O;KA9UX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmUL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnUjB,OAqUO,W;O;KAlVX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuUL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvUjB,OAyUO,W;O;KAtVX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2UL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3UjB,OA6UO,W;O;KA1VX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+UL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/UjB,OAiVO,W;O;KA9VX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmVL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnVjB,OAqVO,W;O;KAlWX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuVL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvVjB,OAyVO,W;O;KAtWX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2VL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3VjB,OA6VO,W;O;KA1WX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+VL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/VjB,OAiWO,W;O;KA9WX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmWL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnWjB,OAqWO,W;O;KAlXX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuWL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvWjB,OAyWO,W;O;KAtXX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2WL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3WjB,OA6WO,W;O;KA1XX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+WL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/WjB,OAiXO,W;O;KA9XX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmXL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnXjB,OAqXO,W;O;KAlYX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuXL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvXjB,OAyXO,W;O;KAtYX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2XL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA3XjB,OA6XO,W;O;KA1YX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA+XL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QA/XjB,OAiYO,W;O;KA9YX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAmYL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAnYjB,OAqYO,W;O;KAlZX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QAuYL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAAY,aAAI,YAAJ,EAAa,OAAb,EAAsB,OAAtB,EAA+B,OAA/B,C;;QAvYjB,OAyYO,W;O;KAtZX,C;6FAgBA,yB;MAAA,wE;MAAA,uC;QAaW,kBAAc,oB;QA2YL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;

AAZ,C;O;MAZ5C,gF;MAAA,yC;QAUI,2D;O;KAVJ,C;IAgBA,sC;MASI,OAAO,yBAAqB,SAArB,EAA2B,SAA3B,C;K;IAGX,4C;MASI,OAAO,gCAA4B,SAA5B,EAAkC,SAAlC,C;K;IAGX,mD;MASI,OAAoD,gBAA7C,gCAA4B,SAA5B,EAAkC,SAAlC,CAA6C,C;K;4GAGxD,yB;MAuNA,wE;MAvNA,oD;QAgOiB,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UAvNsB,U;UAAA,wBAuNT,oBAAmB,cAAnB,EAAmB,sBAAnB,UAAvNS,EAAuNoB,IAAvNpB,WAAW,W;YAA6C,6B;;;QAAhF,OAAgF,W;O;KAAvF,C;8FAAuF,yB;MAAA,wE;MAAA,oD;QAAA,iBAAA,UAAA,C;QAAA,YAAA,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACoC,U;UADpC,YACoC,WADpC,WACoC,WADpC,UACoC,oBADpC,mB,cAAnB,EAAmB,sBAAnB,UAAmB,WAAnB,EAAuNoB,cAAvNpB,C;UAAuN,IAAvNA

AAS,OAAlB,C;UACR,WZ33CG,MAAO,KY23CO,QZ33CP,EY23CiB,CZ33CjB,C;;QY63Cd,OAAO,Q;O;KApB
X,C;6FAuBA,+B;MAWI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,eAAe,SAAS,Q
AAS,OAAlB,C;MACf,OAAO,QAAS,UAAhB,C;QACI,QAAQ,SAAS,QAAS,OAAlB,C;QACR,IAAI,2BAAW,CA
AX,KAAJ,C;UACI,WAAW,C;;;MAGnB,OAAO,Q;K;yFAGX,yB;MAAA,sE;MAAA,kD;QAaI,eAAe,oB;QACf,IA
AI,CAAC,QAAS,UAAd,C;UAAyB,MAAM,6B;QAC/B,eAAe,SAAS,QAAS,OAAlB,C;QACf,OAAO,QAAS,UAA
hB,C;UACI,QAAQ,SAAS,QAAS,OAAlB,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,
CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAtBX,C;qGAyBA,2C;MAWI,eAAe,oB;MACf,IAAI,CAAC,Q
AAS,UAAd,C;QAAyB,OAAO,I;MAChC,eAAe,SAAS,QAAS,OAAlB,C;MACf,OAAO,QAAS,UAAhB,C;QACI,Q
AAQ,SAAS,QAAS,OAAlB,C;QACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;UA
CI,WAAW,C;;;MAGnB,OAAO,Q;K;IAGX,iC;MASI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OA
AO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,MZ18CG,MA
AO,KY08CE,GZ18CF,EY08CO,CZ18CP,C;;MY48Cd,OAAO,G;K;IAGX,iC;MASI,eAAe,oB;MACf,IAAI,CAAC,
QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QA
AS,O;QACjB,MZx+CG,MAAO,KYw+CE,GZx+CF,EYw+CO,CZx+CP,C;;MY0+Cd,OAAO,G;K;IAGX,iC;MAOI
,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAA
S,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAI,sBAAM,CAAN,KAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;
K;IAGX,2C;MAGI,OAAO,4BAAc,UAAd,C;K;IAGX,iD;MAOI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;Q
AAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAA
I,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,
2B;MAII,OAAO,uB;K;IAGX,2B;MAII,OAAO,uB;K;IAGX,2B;MAGI,OAAO,uB;K;iFAGX,+B;MAGW,sB;;QAY
P,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,I
AAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eAAhB,mB,QAAgB,J,CAAS,UAAT,C;;UAEX,Q
AAQ,QAAS,O;UACjB,QAAnBe,QAAmBP,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;UACI,WAAW,C;;;QAED,W;QAClB,qBAAO,O;;;MAzBP,P;K;6FAAJ,+B;MAAA,sE;MAAA,kD;QAaI,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,I;UAAP,uB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAd,C;UAAyB,qBAAO,O;UAAP,uB;;QACzB,eAAhB,mB,QAAgB,J,CAAS,UAAT,C;;UAEX,QA

AAO,QAAS,UAAhB,C;QACI,QAAQ,SAAS,QAAS,OAAlB,C;QACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;UACI,WAAW,C;;;MAGnB,OAAO,Q;K;IAGX,iC;MASI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,MZvjDG,MAAO,KYujDE,GZvjDF,EYujDO,CZvjDP,C;;MYyjDd,OAAO,G;K;IAGX,iC;MASI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,MZrlDG,MAAO,KYqlDE,GZrlDF,EYqlDO,CZrlDP,C;;MYulDd,OAAO,G;K;IAGX,iC;MAOI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAd,C;QAAyB,OAAO,I;MAChC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAI,sBAAM,CAAN,KAAJ,C;UAAa,a,MAAb,E;UAAoB,OAAO,G;K;IATX,sC;MAOhB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;UACI,WAAW,C;;MAGnB,OAAO,G;K;IAGX,4B;MAQI,OAAO,CAAC,oBAAW,U;K;+EAGvB,gC;MAOI,OAAO,Q;MAAQ,2B;MAAA,8B;MAAA,iC;QAAA,OAAA,K;;MAAA,OAAA,I;K;IAGnB,8C

;K;IAXgB,uF;MAAA,yD;uBAAA,8H;YAAA,S;iBAAA,Q;;iBAAA,uB;O;K;IAhBpB,sD;MAgBI,OAAO,SAAS,iDAAT,C;K;IAcX,+C;MAkBI,OAAO,yBAAY,OAAZ,EAAqB,SAArB,C;K;IAGX,sD;MAmBI,OAAO,gCAAmB,OAAnB,EAA4B,SA5B,C;K;gFAGX,+B;MASoB,Q;MADhB,UAAe,C;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,YAAO,SAAS,OAAT,CAAP,I;;MAEJ,OAAO,G;K;4FAGX,+B;MASoB,Q;MADhB,UAAkB,G;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;K;iFAGX,+B;MAYoB,Q;MADhB,UAAoB,C;MACJ,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;K;iFAGX,+B;MAYoB,Q;MADhB,UAAe,C;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,YAAO,SAAS,OAAT,CAAP,I;;MAEJ

B,C;QAAgB,yB;QACZ,IAAI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,CAAR,IAAa,SAAS,KAA1B,C;UACW,gBAAP,MAAO,EAAc,OAAd,EAAuB,SAAvB,C;;UACJ,K;;MAEX,IAAI,SAAS,CAAT,IAAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,MAAO,gBAAO,OAAP,C;MACP,OAAO,M;K;IAGX,4F;MAUwC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAA,a,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAoC,I;MAAjN,cAAO,OAAO,oBAAO,sBAAP,EAAwB,EAAxB,wB,SAAwB,cAAxB,C,SAAwB,cAAxB,C,SAAwB,cAAxB,C

Ab,E;O;KAPjE,C;gGAUA,yB;MAAA,+C;MAAA,mC;QAOI,OAAY,UAAL,SAAK,EAAU,KAAV,C;O;KAPhB,C;0EAUA,yB;MA4EA,6C;MAAA,oC;MAAA,gC;MA5EA,uC;QAOW,sB;;UAyES,Q;UAAA,0B;UAAhB,OAAgB,cAAhB,C;YAAgB,oC;YAAM,IAzEH,SAyEO,CAAU,oBAAV,CAAJ,C;cAAwB,qBAAO,O;cAAP,uB;;;UAC9C,qBAAO,I;;;QA1EP,yB;O;KAPJ,C;kFAUA,yB;MAwJA,mD;MAAA,+C;MAAA,oC;MAxJA,uC;QAOW,qB;;UAuJO,Q;UAAA,OAAa,SAAR,sBAAQ,CAAb,W;UAAd,OAAc,cAAd,C;YAAc,uB;YACV,cAAc,qBAAK,KAAL,C;YACd,IAzJc,SAyJV,CAAU,oBAAV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;;QA3JP,3J,wB;O;KAPJ,C;kFAUA,yB;MAiKA,6B;MAAA,oC;MAAA,oC;MAAA,gC;MAAA,iE;MAAA,uC;QAOW,oBAAK,Q;QAAL,OAAa,O;;QAAb,qCAAa,MAAb,gCAAuB,B,6DAAvB,B,C;O;KAPX,C;kFAUA,yB

I,ckBwTc,wD;QlBvTd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MkBwTV,OAAO,mBAAkB,gBAAZ,mBAAS,CAAT,IAAY,EAAc,CAAd,CAAlB,C;K;IAGX,mC;MlB7TI,IAAI,EkBqUI,KAAK,ClBrUT,CAAJ,C;QACI,ckBoUc,wD;QlBnUd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MkBoUV,OAAO,mBAAkB,gBAAZ,mBAAS,CAAT,IAAY,EAAc,CAAd,CAAlB,C;K;2FAGX,yB;MAAA,uD;MAAA,oC;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAY,CAAL,C;YACI,OAAO,8BAAY,CAAZ,EAAe,QAAQ,CAAR,IAAf,C;QACf,OAAO,E;O;KATX,C;4FAYA

Q,WAAjC,C;;MkByiBV,aAAa,gB;MACb,OL9W6E,oBK8W5D,SAAW,eAAF,CAAE,EAAa,MAAb,CAAX,IL9W4D,C;K;2FKiXjF,yB;MAAA,uD;MAAA,oC;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAV,CAAL,C;YACI,OAAO,8BAAY,QAAQ,CAAR,IAAZ,EAAuB,gBAAvB,C;;;QAGf,OAAO,8BAAY,CAAZ,EAAe,gBAAf,C;O;KAXX,C;4FAcA,yB;MAAA,uD;MAAA,oC;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAV,CAAL,C;YACI,OLvYqE,oBKuYpD,QAAQ,CAAR,IAAQ,CAAR,EAAW,CAAX,C;;;QK0Y7E,OAAO,S;O;KAXX,C;oFAcA,yB;MAAA,uC;MAAA,uC;QAM0B,QAAM0B,Q;QAAA,uB;QAAtB,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAV,CAAL,C;YACI,OAAO,8BAAY,QAAQ,CAAR,IAAZ,EAAuB,gBAAvB,C;;;QAGf,OAAO,8BAAY,CAAZ,EAAe,gBAAf,C;O;KAXX,C;oFAcA,yB;MAAA,uC;MAAA,uC;QAM0B,QAAM0B,Q;QAAA,uB;QAAtB,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAV,CAAL,C;YACI,OLvYqE,oBKuYpD,QAAQ,CAAR,IAAQ,CAAR,EAAW,CAAX,C;;;QK0Y7E,OAAO,S

C;MAAA,oD;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,WAAW,UAAU,oBAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAEhB,OAAO,W;O;KARX,C;gFAWA,yB;MAAA,wE;MAyBA,6C;MAAA,oC;MAAA,+D;MAAA,gC;MAzBA,yC;QASW,kBAAU,oB;QAyBD,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,UA1BoD,WA0B1C,CAAY,oBAAZ,C;UzBrjBP,U;UADP,YyBujBe,WzBvjBH,WyBujBwB,GzBvjBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBqjBuC,gB;YAA5B,WzBpjBX,ayBojBgC,GzBpjBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBijBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QA5BT,OA8BO,W;O;KAvCX,C;gFAYA,yB;MAAA,wE;MA8BA,6C;MAAA,oC;MAAA,+D;MAAA,gC;MA9BA,yD;QAUW,kBAAU,oB;QA8BD,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,UA/BA,/BiD,WA+BA,CAAY,oBAAZ,C;UzBvkBP,U;UADP,YyBykBe,WzBzkBH,WyBykBwB,GzBzkBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBukBuC,gB;YAA5B,WzBtkBX,ayBskBgC,GzBtkBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBmkBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAhCX,C;oFAmCA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBzlBO,U;UADP,YyB2lBe,WzB3lBH,WyB2lBwB,GzB3lBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBylBuC,gB;YAA5B,WzBxlBX,ayBwlBgC,GzBxlBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBqlBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzB7mBO,U;UADP,YyB+mBe,WzB/mBH,WyB+mBwB,GzB/mBxB,C;UACL,IAAI,aAAJ,C;YACH,ayB6mBuC,gB;YAA5B,WzB5mBX,ayB4mBgC,GzB5mBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBymBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjoBO,U;UADP,YyBmoBe,WzBnoBH,WyBmoBwB,GzBnoBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBioBuC,gB;YAA5B,WzBhoBX,ayBgoBgC,GzBhoBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6nBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrpBO,U;UADP,YyBupBe,WzBvpBH,WyBupBwB,GzBvpBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBqpBuC,gB;YAA5B,WzBppBX,ayBopBgC,GzBppBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBipBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBzqBO,U;UADP,YyB2qBe,WzB3qBH,WyB2qBwB,GzB3qBxB,C;UACL,IAAI,aAAJ,C;YACH,ayByqBuC,gB;YAA5B,WzBxqBX,ayBwqBgC,GzBxqBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBqqBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjsBO,U;UADP,YyBmsBe,WzBnsBH,WyBmsBwB,GzBnsBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBisBuC,gB;YAA5B,WzBhsBX,ayBgsBgC,GzBhsBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6rBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrtBO,U;UADP,YyButBe,WzBvtBH,WyButBwB,GzBvtBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBqtBuC,gB;YAA5B,WzBptBX,ayBotBgC,GzBptBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBitBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBzuBO,U;UADP,YyB2uBe,WzB3uBH,WyB2uBwB,GzB3uBxB,C;UACL,IAAI,aAAJ,C;YACH,ayByuBuC,gB;YAA5B,WzBxuBX,ayBwuBgC,GzBxuBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBquBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjvBO,U;UADP,YyBmvBe,WzBnvBH,WyBmvBwB,GzBnvBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBivBuC,gB;YAA5B,WzBhvBX,ayBgvBgC,GzBhvBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6uBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrwBO,U;UADP,YyBuwBe,WzBvwBH,WyBuwBwB,GzBvwBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBqwBuC,gB;YAA5B,WzBpwBX,ayBowBgC,GzBpwBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBiwBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBzxBO,U;UADP,YyB2xBe,WzB3xBH,WyB2xBwB,GzB3xBxB,C;UACL,IAAI,aAAJ,C;YACH,ayByxBuC,gB;YAA5B,WzBxxBX,ayBwxBgC,GzBxxBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBqxBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjyBO,U;UADP,YyBmyBe,WzBnyBH,WyBmyBwB,GzBnyBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBiyBuC,gB;YAA5B,WzBhyBX,ayBgyBgC,GzBhyBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6xBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrzBO,U;UADP,YyBuzBe,WzBvzBH,WyBuzBwB,GzBvzBxB,C;UACL,IAAI,aAAJ,C;YACH,ayBqzBuC,gB;YAA5B,WzBpzBX,ayBozBgC,GzBpzBhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBizBA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBz0BO,U;UADP,YyB20BE,WzB30BU,WyB20BiB,GzB30BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBy0BuC,gB;YAA5B,WzBx0BX,ayBw0BgC,GzBx0BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBq0BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBj1BO,U;UADP,YyBm1BE,WzBn1BU,WyBm1BiB,GzBn1BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBi1BuC,gB;YAA5B,WzBh1BX,ayBg1BgC,GzBh1BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB60BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBr2BO,U;UADP,YyBu2BE,WzBv2BU,WyBu2BiB,GzBv2BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBq2BuC,gB;YAA5B,WzBp2BX,ayBo2BgC,GzBp2BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBi2BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBz3BO,U;UADP,YyB23BE,WzB33BU,WyB23BiB,GzB33BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBy3BuC,gB;YAA5B,WzBx3BX,ayBw3BgC,GzBx3BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBq3BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBj4BO,U;UADP,YyBm4BE,WzBn4BU,WyBm4BiB,GzBn4BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBi4BuC,gB;YAA5B,WzBh4BX,ayBg4BgC,GzBh4BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB63BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBr5BO,U;UADP,YyBu5BE,WzBv5BU,WyBu5BiB,GzBv5BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBq5BuC,gB;YAA5B,WzBp5BX,ayBo5BgC,GzBp5BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBi5BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBz6BO,U;UADP,YyB26BE,WzB36BU,WyB26BiB,GzB36BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBy6BuC,gB;YAA5B,WzBx6BX,ayBw6BgC,GzBx6BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBq6BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBj7BO,U;UADP,YyBm7BE,WzBn7BU,WyBm7BiB,GzBn7BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBi7BuC,gB;YAA5B,WzBh7BX,ayBg7BgC,GzBh7BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB66BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBr8BO,U;UADP,YyBu8BE,WzBv8BU,WyBu8BiB,GzBv8BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBq8BuC,gB;YAA5B,WzBp8BX,ayBo8BgC,GzBp8BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBi8BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBz9BO,U;UADP,YyB29BE,WzB39BU,WyB29BiB,GzB39BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBy9BuC,gB;YAA5B,WzBx9BX,ayBw9BgC,GzBx9BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBq9BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBj+BO,U;UADP,YyBm+BE,WzBn+BU,WyBm+BiB,GzBn+BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBi+BuC,gB;YAA5B,WzBh+BX,ayBg+BgC,GzBh+BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB69BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBr/BO,U;UADP,YyBu/BE,WzBv/BU,WyBu/BiB,GzBv/BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBq/BuC,gB;YAA5B,WzBp/BX,ayBo/BgC,GzBp/BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBi/BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBz/BO,U;UADP,YyB2/BE,WzB3/BU,WyB2/BiB,GzB3/BjB,C;UACL,IAAI,aAAJ,C;YACH,ayBy/BuC,gB;YAA5B,WzBx/BX,ayBw/BgC,GzBx/BhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBq/BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjgCO,U;UADP,YyBmgCE,WzBngCU,WyBmgCiB,GzBngCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBigCuC,gB;YAA5B,WzBhgCX,ayBggCgC,GzBhgChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6/BA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrhCO,U;UADP,YyBuhCE,WzBvhCU,WyBuhCiB,GzBvhCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBqhCuC,gB;YAA5B,WzBphCX,ayBohCgC,GzBphChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBihCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBziCO,U;UADP,YyB2iCE,WzB3iCU,WyB2iCiB,GzB3iCjB,C;UACL,IAAI,aAAJ,C;YACH,ayByiCuC,gB;YAA5B,WzBxiCX,ayBwiCgC,GzBxiChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBqiCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjjCO,U;UADP,YyBmjCE,WzBnjCU,WyBmjCiB,GzBnjCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBijCuC,gB;YAA5B,WzBhjCX,ayBgjCgC,GzBhjChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6iCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrkCO,U;UADP,YyBukCE,WzBvkCU,WyBukCiB,GzBvkCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBqkCuC,gB;YAA5B,WzBpkCX,ayBokCgC,GzBpkChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBikCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBzlCO,U;UADP,YyB2lCE,WzB3lCU,WyB2lCiB,GzB3lCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBylCuC,gB;YAA5B,WzBxlCX,ayBwlCgC,GzBxlChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBqlCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjmCO,U;UADP,YyBmmCE,WzBnmCU,WyBmmCiB,GzBnmCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBimCuC,gB;YAA5B,WzBhmCX,ayBgmCgC,GzBhmChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6lCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrnCO,U;UADP,YyBunCE,WzBvnCU,WyBunCiB,GzBvnCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBqnCuC,gB;YAA5B,WzBpnCX,ayBonCgC,GzBpnChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBinCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBzoCO,U;UADP,YyB2oCE,WzB3oCU,WyB2oCiB,GzB3oCjB,C;UACL,IAAI,aAAJ,C;YACH,ayByoCuC,gB;YAA5B,WzBxoCX,ayBwoCgC,GzBxoChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBqoCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjpCO,U;UADP,YyBmpCE,WzBnpCU,WyBmpCiB,GzBnpCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBipCuC,gB;YAA5B,WzBhpCX,ayBgpCgC,GzBhpChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6oCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrqCO,U;UADP,YyBuqCE,WzBvqCU,WyBuqCiB,GzBvqCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBqqCuC,gB;YAA5B,WzBpqCX,ayBoqCgC,GzBpqChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBiqCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBzrCO,U;UADP,YyB2rCE,WzB3rCU,WyB2rCiB,GzB3rCjB,C;UACL,IAAI,aAAJ,C;YACH,ayByrCuC,gB;YAA5B,WzBxrCX,ayBwrCgC,GzBxrChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBqrCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjsCO,U;UADP,YyBmsCE,WzBnsCU,WyBmsCiB,GzBnsCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBisCuC,gB;YAA5B,WzBhsCX,ayBgsCgC,GzBhsChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6rCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrtCO,U;UADP,YyButCE,WzBvtCU,WyButCiB,GzBvtCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBqtCuC,gB;YAA5B,WzBptCX,ayBotCgC,GzBptChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBitCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBzuCO,U;UADP,YyB2uCE,WzB3uCU,WyB2uCiB,GzB3uCjB,C;UACL,IAAI,aAAJ,C;YACH,ayByuCuC,gB;YAA5B,WzBxuCX,ayBwuCgC,GzBxuChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBquCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjvCO,U;UADP,YyBmvCE,WzBnvCU,WyBmvCiB,GzBnvCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBivCuC,gB;YAA5B,WzBhvCX,ayBgvCgC,GzBhvChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6uCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrwCO,U;UADP,YyBuwCE,WzBvwCU,WyBuwCiB,GzBvwCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBqwCuC,gB;YAA5B,WzBpwCX,ayBowCgC,GzBpwChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBiwCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBzxCO,U;UADP,YyB2xCE,WzB3xCU,WyB2xCiB,GzB3xCjB,C;UACL,IAAI,aAAJ,C;YACH,ayByxCuC,gB;YAA5B,WzBxxCX,ayBwxCgC,GzBxxChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBqxCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBjyCO,U;UADP,YyBmyCE,WzBnyCU,WyBmyCiB,GzBnyCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBiyCuC,gB;YAA5B,WzBhyCX,ayBgyCgC,GzBhyChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyB6xCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBrzCO,U;UADP,YyBuzCE,WzBvzCU,WyBuzCiB,GzBvzCjB,C;UACL,IAAI,aAAJ,C;YACH,ayBqzCuC,gB;YAA5B,WzBpzCX,ayBozCgC,GzBpzChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBizCA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX,C;oFA0CA,yB;MAAA,4E;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,yD;QAAA,kBAAA,oB;QAAA,Q;QAAA,0B;QAAA,OAAA,cAAA,C;UAAA,oC;UACA,UAAA,YAAA,oBAAA,C;UzBz0CO,U;UADP,YyB20CE,WzB30CU,WyB20CiB,GzB30CjB,C;UACL,IAAI,aAAJ,C;YACH,ayBy0CuC,gB;YAA5B,WzBx0CX,ayBw0CgC,GzBx0ChC,EAAS,MAAT,C;YACA,e;;YAEA,c;;UyBq0CA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;O;KAvCX

QAF9C,YAAY,C;QACZ,kBAAkB,O;QACF,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QACpC,OAAO,W;O;KAbX,C;mFAgBA,yB;MAAA,uD;MAAA,oC;MAAA,gD;QAYoC,Q;QAHhC,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,kCAAI,YAAJ,EAAI,oBAAJ,SAAa,EAAb,wB,WAAxB,C;;QAElB,OAAO,W;O;KAdX,C;iGAiBA,yB;MAAA,uD;MAAA,oC;MAAA,gD;QAUI,YAAY,wB;QACZ,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAU,iCAAI,KAAJ,EAAI,EAAjB,iB,iCAAiB,KAAjB,EAAiB,EAAjB,6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;O;KAhBX,C;gFAmBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAvBX,C;8FA0BA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA9BX,C;8FAiCA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KArCX,C;8FAwCA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA5CX,C;8FA+CA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAnDX,C;8FAsDA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA1DX,C;8FA6DA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAjEX,C;8FAoEA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAxEX,C;8FA2EA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA/EX,C;8FAkFA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAtFX,C;8FAyFA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA7FX,C;8FAgGA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KApGX,C;8FAuGA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA3GX,C;8FA8GA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAlHX,C;8FAqHA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAzHX,C;8FA4HA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAhIX,C;8FAmIA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAvIX,C;8FA0IA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA9IX,C;8FAiJA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KArJX,C;8FAwJA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA5JX,C;8FA+JA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAnKX,C;8FAsKA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA1KX,C;8FA6KA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAjLX,C;8FAoLA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAxLX,C;8FA2LA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA/LX,C;8FAkMA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAtMX,C;8FAyMA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA7MX,C;8FAgNA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KApNX,C;8FAuNA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA3NX,C;8FA8NA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAlOX,C;8FAqOA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAzOX,C;8FA4OA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAhPX,C;8FAmPA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAvPX,C;8FA0PA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA9PX,C;8FAiQA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KArQX,C;8FAwQA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA5QX,C;8FA+QA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAnRX,C;8FAsRA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA1RX,C;8FA6RA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAjSX,C;8FAoSA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAxSX,C;8FA2SA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA/SX,C;8FAkTA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAtTX,C;8FAyTA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA7TX,C;8FAgUA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KApUX,C;8FAuUA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA3UX,C;8FA8UA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAlVX,C;8FAqVA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAzVX,C;8FA4VA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAhWX,C;8FAmWA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAvWX,C;8FA0WA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA9WX,C;8FAiXA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KArXX,C;8FAwXA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA5XX,C;8FA+XA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAnYX,C;8FAsYA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA1YX,C;8FA6YA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAjZX,C;8FAoZA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAxZX,C;8FA2ZA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA/ZX,C;8FAkaA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAtaX,C;8FAyaA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA7aX,C;8FAgbA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KApbX,C;8FAubA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA3bX,C;8FA8bA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAlcX,C;8FAqcA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAzcX,C;8FA4cA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAhdX,C;8FAmdA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAvdX,C;8FA0dA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA9dX,C;8FAieA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KArfX,C;8FAwfA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA5fX,C;8FA+fA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAngBX,C;8FAsgBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA1gBX,C;8FA6gBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAjhBX,C;8FAohBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAxhBX,C;8FA2hBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA/hBX,C;8FAkiBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAtiBX,C;8FAyiBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA7iBX,C;8FAgjBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KApjBX,C;8FAujBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA3jBX,C;8FA8jBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAlkBX,C;8FAqkBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAzkBX,C;8FA4kBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAhlBX,C;8FAmlBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAvlBX,C;8FA0lBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA9lBX,C;8FAimBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KArmBX,C;8FAwmBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA5mBX,C;8FA+mBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAnnBX,C;8FAsnBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA1nBX,C;8FA6nBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAjoBX,C;8FAooBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAxoBX,C;8FA2oBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KA/oBX,C;8FAkpBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAhC,C;;QAEpC,OAAO,W;O;KAtpBX,C;8FAypBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,oBAAh

AAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,2B;MAGI,OAAO,uB;K;4EAGX,yB;MAMA,uD;MAAA,oC;MANA,sC;QAGW,sB;;UAUP,ICtnCgD,qBAAU,CDsnC1D,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,qBAAK,CAAL,C;UACd,gBAAqB,wB;UACrB,IAAI,cAAa,CAAjB,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eAAmB,QAAcJ,CAAS,oBAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,qBAAK,CAAL,C;YACR,QAAjBe,QAAiBP,P,CAAS,cAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QAAvBP,P,yB;O;KAHJ,C;wFAMA,yB;MAAA,uD;MAAA,oC;MAAA,sC;QAOI,ICtnCgD,qBAAU,CDsnC1D,C;UAAe,OAAO,I;QACtB,cAAc,qBAAK,CAAL,C;QAAd,gBAAqB,cAAL,SAAK,C;QAArB,IAAI,cAAa,CAAjB,C;UAAoB,qBAAO,O;UAAP,uB;;QACpB,eAAmB,QAAcJ,CAAS,oBAAT,C;QAAjC,IAAI,cAAa,CAAjB,C;UAAoB,qBAAO,O;UAAP,uB;;QACpB

O;KARJ,C;8EAWA,yB;MAAA,4F;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,ICn4CgD,qBAAU,CDm4C1D,C;UACI,MAAM,mCAA8B,uCAA9B,C;QACV,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,wBAAV,EAAuB,iCAAK,KAAL,EAAvB,E;;QAElB,OAAO,W;O;KAnBX,C;4FAsBA,yB;MAAA,4F;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,ICz5CgD,qBAAU,CDy5C1D,C;UACI,MAAM,mCAA8B,uCAA9B,C;QACV,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;wGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/6CgD,qBAAU,CD+6C1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;0FAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,ICt8CgD,qBAAU,CDs8C1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,wBAAV,EAAuB,iCAAK,KAAL,EAAvB,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/9CgD,qBAAU,CD+9C1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/gDgD,qBAAU,CD+gD1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/jDgD,qBAAU,CD+jD1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/mDgD,qBAAU,CD+mD1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/pDgD,qBAAU,CD+pD1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/sDgD,qBAAU,CD+sD1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/vDgD,qBAAU,CD+vD1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/yDgD,qBAAU,CD+yD1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/1DgD,qBAAU,CD+1D1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/4DgD,qBAAU,CD+4D1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/7DgD,qBAAU,CD+7D1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/+DgD,qBAAU,CD++D1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/iEgD,qBAAU,CD+iE1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/lEgD,qBAAU,CD+lE1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/oEgD,qBAAU,CD+oE1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/rEgD,qBAAU,CD+rE1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/uEgD,qBAAU,CD+uE1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/xEgD,qBAAU,CD+xE1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/0EgD,qBAAU,CD+0E1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/3EgD,qBAAU,CD+3E1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/6EgD,qBAAU,CD+6E1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/9EgD,qBAAU,CD+9E1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/gFgD,qBAAU,CD+gF1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/jFgD,qBAAU,CD+jF1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/mFgD,qBAAU,CD+mF1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/pFgD,qBAAU,CD+pF1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/sFgD,qBAAU,CD+sF1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/vFgD,qBAAU,CD+vF1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/yFgD,qBAAU,CD+yF1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/1FgD,qBAAU,CD+1F1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/4FgD,qBAAU,CD+4F1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/7FgD,qBAAU,CD+7F1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/+FgD,qBAAU,CD++F1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/iGgD,qBAAU,CD+iG1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/lGgD,qBAAU,CD+lG1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/oGgD,qBAAU,CD+oG1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/rGgD,qBAAU,CD+rG1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/uGgD,qBAAU,CD+uG1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/xGgD,qBAAU,CD+xG1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/0GgD,qBAAU,CD+0G1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/3GgD,qBAAU,CD+3G1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/6GgD,qBAAU,CD+6G1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/9GgD,qBAAU,CD+9G1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/gHgD,qBAAU,CD+gH1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/jHgD,qBAAU,CD+jH1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/mHgD,qBAAU,CD+mH1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/pHgD,qBAAU,CD+pH1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/sHgD,qBAAU,CD+sH1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/vHgD,qBAAU,CD+vH1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/yHgD,qBAAU,CD+yH1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/1HgD,qBAAU,CD+1H1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/4HgD,qBAAU,CD+4H1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/7HgD,qBAAU,CD+7H1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/+HgD,qBAAU,CD++H1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/iIgD,qBAAU,CD+iI1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/lIgD,qBAAU,CD+lI1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/oIgD,qBAAU,CD+oI1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/rIgD,qBAAU,CD+rI1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/uIgD,qBAAU,CD+uI1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/xIgD,qBAAU,CD+xI1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/0IgD,qBAAU,CD+0I1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/3IgD,qBAAU,CD+3I1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/6IgD,qBAAU,CD+6I1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/9IgD,qBAAU,CD+9I1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/gJgD,qBAAU,CD+gJ1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/jJgD,qBAAU,CD+jJ1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB,OAAO,W;O;KAnBX,C;uGAsBA,yB;MAAA,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAgBqB,Q;QAHjB,IC/mJgD,qBAAU,CD+mJ1D,C;UACI,OAAO,I;QACX,kBAAkB,qBAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAElB

,EAA8B,iCAAK,KAAL,EAA9B,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,4BAAO,M;;;QAkFP,gC;O;KAfJ,C;4EAkBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,sC;QAOoB,Q;QADhB,UAAe,C;QACC,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,YAAO,SAAS,oBAAT,CAAP,I;;QAEJ,OAAO,G;O;KAVX,C;wFAaA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,sC;QAOoB,Q;QADhB,UAAkB,G;QACF,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,OAAO,SAAS,oBAAT,C;;QAEX,OAAO,G;O;KAVX,C;4EAaA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,sC;QAUoB,Q;QADhB,UAAoB,C;QACJ,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,OAAO,SAAS,oBAAT,C;;QAEX,OAAO,G;O;KAVX,C;4EAgBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,sC;QAUoB,Q;QADhB,UAAe,C;QACC,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,YAAO,SAAS,oBAAT,CAAP,I;;QAEJ,OAAO,G;O;KAVX,C;4EAgBA,yB;MAAA,SASoB,gB;MATpB,6C;MAAA,oC;MAAA,gC;MAAA,sC;QAUoB,Q;QADhB,UAAe,Y;QACC,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,cAAO,SAAS,oBAAT,CAAP,C;;QAEJ,OAAO,G;O;KAVX,C;4EAgBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,sC;QAUoB,Q;QADhB,UAAkB,C;QACF,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,OAAO,SAAS,oBAAT,C;;QAEX

MAAA,qD;MAAA,gE;MAAA,oC;MAAA,uC;QAUI,WAAW,mBAAS,CAAT,I;QACX,IAAI,OAAO,CAAX,C;UA
Ac,OAAO,W;QACrB,aAAa,eAAa,IAAb,C;QACb,iBAAc,CAAd,UAAsB,IAAtB,U;UACI,MAAO,WAAI,UAAU,i
CAAK,KAAL,EAAV,EAAuB,iCAAK,QAAQ,CAAR,IAAL,EAAvB,CAAJ,C;;QAEX,OAAO,M;O;KAhBX,C;IA
wBoB,8C;MAAA,mB;QAAE,OAAK,WAAL,eAAK,C;O;K;IAL3B,kC;MAIQ,wC;MAAA,S;QAAkB,OCniE0B,qB
AAU,C;;MDmiE1D,S;QAAiC,OAAO,W;MACxC,oCAAgB,8BAAhB,C;K;IAQgB,8C;MAAA,mB;QAAE,OAAK,
WAAL,eAAK,C;O;K;IAL3B,kC;MAIQ,wC;MAAA,S;QAAkB,OC3iE0B,qBAAU,C;;MD2iE1D,S;QAAiC,OAAO,
e;MACxC,oCAAgB,8BAAhB,C;K;IEpwEkC,yC;MAAA,wB;QAAW,OAAA,aAAK,KAAL,ChCsLV,K;O;K;IiCtL
H,wC;MAAA,wB;QAAW,OAAA,aAAK,KAAL,ChC8NV,K;O;K;IiC9NC,yC;MAAA,wB;QAAW,OAAA,aAAK,
KAAL,CjByOV,K;O;K;IkBzOC,0C;MAAA,wB;QAAW,OAAA,aAAK,KAAL,CjCiMV,K;O;K;4FkC5PzC,qB;MA
UI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,CA
AJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;4FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;M
AUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,C
AAJ,C;K;4FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;
MAUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;4FAGX,qB;MAUI,OAAO,sBAAI,
CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,q
B;MAUI,OAAO,sBAAI,CAAJ,C;K;4FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBA
AI,CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;6FAGX,qB;MAUI,OAAO,sBAAI,CAAJ,C;K;uGAuC
X,yB;MA8gHI,8D;MA9gHJ,iD;QASe,oBAAS,C;QAAT,S;UAAc,gBAqgHT,cAAR,iBAAQ,C;;QArgHhB,OAAO,
OAAsC,sBAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KATjE,C;uGAYA,yB;MA0gHI,8D;MA1gHJ,iD;QASe,o
BAAS,C;QAAT,S;UAAc,gBAigHT,cAAR,iBAAQ,C;;QAjgHhB,OAAO,OAAsC,sBAAI,KAAJ,CAAtC,GAAsD,a
AAa,KAAb,C;O;KATjE,C;uGAYA,yB;MAsgHI,8D;MAtgHJ,iD;QASe,oBAAS,C;QAAT,S;UAAc,gBA6/GT,cAA
R,iBAAQ,C;;QA7/GhB,OAAO,OAAsC,sBAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KATjE,C;uGAYA,yB;M
AkgHI,8D;MAlgHJ,iD;QASe,oBAAS,C;QAAT,S;UAAc,gBAy/GT,cAAR,iBAAQ,C;;QAz/GhB,OAAO,OAAsC,s
BAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;O;KATjE,C;uGAYA,yB;MAAA,sD;MAAA,mC;QASI,OAAY,UAA
L,SAAK,EAAU,KAAV,C;O;KAThB,C;uGAYA,yB;MAAA,sD;MAAA,mC;QASI,OAAY,UAAL,SAAK,EAAU,K
AAV,C;O;KAThB,C;uGAYA,yB;MAAA,sD;MAAA,mC;QASI,OAAY,UAAL,SAAK,EAAU,KAAV,C;O;KAThB
,C;uGAYA,yB;MAAA,sD;MAAA,mC;QASI,OAAY,UAAL,SAAK,EAAU,KAAV,C;O;KAThB,C;iFAYA,gC;MA
SW,sB;;QA8NS,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IA9NH,SA8NO,CAAU,OAAV,CA
AJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;MA/NP,yB;K;iFAGJ,gC;MASW,sB;;QA6NS,Q;QAAA,
2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IA7NH,SA6NO,CAAU,OAAV,CAAJ,C;YAAwB,qBAAO,O;
YAAP,uB;;;QAC9C,qBAAO,I;;;MA9NP,yB;K;iFAGJ,gC;MASW,sB;;QA4NS,Q;QAAA,2B;QAAhB,OAAgB,cAA
hB,C;UAAgB,yB;UAAM,IA5NH,SA4NO,CAAU,OAAV,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBA
AO,I;;;MA7NP,yB;K;iFAGJ,gC;MASW,sB;;QA2NS,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAA
M,IA3NH,SA2NO,CAAU,OAAV,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;MA5NP,yB;K;yF
AGJ,yB;MA4nBA,+C;MAkuFI,0D;MA91GJ,uC;QASW,qB;;UA4nBO,Q;UAAA,OAAa,SAytFX,YAAR,iBAAQ,C
AztFW,CAAb,W;UAAd,OAAc,cAAd,C;YAAc,uB;YACV,cAAc,sBAAK,KAAL,C;YACd,IA9nBc,SA8nBV,CAA
U,OAAV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;;QAhoBP,wB;O;KATJ,C;yFAYA,yB;MAgo
BA,+C;MA0tFI,0D;MA11GJ,uC;QASW,qB;;UAgoBO,Q;UAAA,OAAa,SAitFX,YAAR,iBAAQ,CAjtFW,CAAb,
W;UAAd,OAAc,cAAd,C;YAAc,uB;YACV,cAAc,sBAAK,KAAL,C;YACd,IAloBc,SAkoBV,CAAU,OAAV,CAAJ
,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;;QApoBP,wB;O;KATJ,C;yFAYA,yB;MAooBA,+C;MAktFI,
0D;MAt1GJ,uC;QASW,qB;;UAooBO,Q;UAAA,OAAa,SAysFX,YAAR,iBAAQ,CAzsFW,CAAb,W;UAAd,OAAc,
cAAd,C;YAAc,uB;YACV,cAAc,sBAAK,KAAL,C;YACd,IAtoBc,SAsoBV,CAAU,OAAV,CAAJ,C;cAAwB,oBA
AO,O;cAAP,sB;;;UAE5B,oBAAO,I;;;QAxoBP,wB;O;KATJ,C;yFAYA,yB;MAwoBA,+C;MA0sFI,0D;MAl1GJ,uC
;QASW,qB;;UAwoBO,Q;UAAA,OAAa,SAisFX,YAAR,iBAAQ,CAjsFW,CAAb,W;UAAd,OAAc,cAAd,C;YAAc,
uB;YACV,cAAc,sBAAK,KAAL,C;YACd,IA1oBc,SA0oBV,CAAU,OAAV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB
;;;UAE5B,oBAAO,I;;;QA5oBP,wB;O;KATJ,C;mFAYA,yB;MAAA,8C;MnCpHA,6B;MmCoHA,4B;QAQI,OnClH
mC,cmCkHpB,MAAR,iBAAQ,CnClHoB,C;O;KmC0GvC,C;mFAWA,yB;MAAA,8C;MnBhHA,+B;MmBgHA,4B;
QAQI,OnB9GsC,emB8GvB,MAAR,iBAAQ,CnB9GuB,C;O;KmBsG1C,C;mFAWA,yB;MAAA,8C;MpCxLA,+B;

MoCwLA,4B;QAQI,OpCtLsC,eoCsLvB,MAAR,iBAAQ,CpCtLuB,C;O;KoC8K1C,C;mFAWA,yB;MAAA,8C;MlCtLA,iC;MkCsLA,4B;QAQI,OlCpLyC,gBkCoL1B,MAAR,iBAAQ,ClCpL0B,C;O;KkC4K7C,C;mFAWA,yB;MAAA,iE;MAAA,uC;QAQoB,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;O;KATV,C;mFAY

mB,kBAAR,iB;QAAQ,sB;;UtCg5BD,Q;UAAA,OAAQ,SAAR,wBAAQ,CAAR,W;UAAd,OAAc,cAAd,C;YAAc,uB;YACV,IsCj5ByB,UnCjcM,cHk1CjB,YAAK,KAAL,CGl1CiB,CmCicN,CtCi5BzB,C;cACI,qBAAO,K;cAAP,uB;;;UAGR,qBAAO,E;;;QsCr5BP,yB;O;KAPJ,C;+FAUA,yB;MAAA,sC;MtCq5BA,0D;MAAA,+C;MmBp1CA,+B;MmB+bA,yBAO+B,yB;QnBtc/B,+B;emBsc+B,6B;UAAA,qB;YAAE,yBnB5bS,emB4bC,EnB5bD,CmB4bT,C;W;S;OAAF,C;MAP/B,uC;QAOmB,kBAAR,iB;QAAQ,sB;;UtCk5BD,Q;UAAA,OAAQ,SAAR,wBAAQ,CAAR,W;UAAd,OAAc,cAAd,C;YAAc,uB;YACV,IsCn5ByB,UnB5bS,enB+0CpB,YAAK,KAAL,CmB/0CoB,CmB4bT,CtCm5BzB,C;cACI,qBAAO,K;cAAP,uB;;;UAGR,qBAAO,E;;;QsCv5BP,yB;O;KAPJ,C;+FAUA,yB;MAAA,sC;MtCu2BA,0D;MAAA,+C;ME72CA,+B;MoCsgBA,yBAO+B,yB;QpC7gB/B,+B;eoC6gB+B,6B;UAAA,qB;YAAE,yBnBpgBS,eoCmgBC,EpCngBD,CmBogBT,C;W;S;OAAF,C;MAP/B,uC;QAOmB,kBAAR,iB;QAAQ,sB;;UtCo2BD,Q;UAAA,OAAQ,SAAR,wBAAQ,CAAR,W;UAAd,OAAc,cAAd,C;YAAc,uB;YACV,IsCr2ByB,UpCngBS,eFw2CpB,YAAK,KAAL,CEx2CoB,CoCmgBT,CtCq2BzB,C;cACI,qBAAO,K;cAAP,uB;;;UAGR,qBAAO,E;;;QsCz2BP,yB;O;KAPJ,C;+FAUA,yB;MAAA,sC;MtCy0BA,0D;MAAA,+C;MI52CA,iC;MkCmgBA,yBAO+B,yB;QlC1gB/B,+B;ekC0gB+B,6B;UAAA,qB;YAAE,yBnBjgBS,eoCggBC,EpCngBD,CmBogBT,C;W;S;OAAF,C;MAP/B,uC;QAOmB,kBAAR,iB;QAAQ,sB;;UtCs0BD,Q;UAAA,OAAQ,SAAR,wBAAQ,CAAR,W;UAAd,OAAc,cAAd,C;YAAc,uB;YACV,IsCv0ByB,UlCngBS,eFw2CpB,YAAK,KAAL,CEx2CoB,CkCmgBT,CtCu0BzB,C;cACI,qBAAO,K;cAAP,uB;;;UAGR,qBAAO,E;;;QsC30BP,yB;O;KAPJ,C;+FAUA,yB;MAAA,sC;MtC20

AgBA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QASI,OAAO,kBAAO,cAAP,C;O;KATX,C;qFAYA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QASI,OAAO,kBAAO,cAAP,C;O;KATX,C;qFAYA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QASI,OAAO,kBAAO,cAAP,C;O;KATX,C;qFAYA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QASI,OAAO,kBAAO,cAAP,C;O;KATX,C;IAYA,sC;MAQI,IAAI,mBAAJ,C;QACI,MAAM,2BAAuB,iBAAAvB,C;MACV,OAAO,sBAAAI,MAAO,iBAAAQ,Q,iBAAAA,cAAP,C;K;IAGX

MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB8hDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAApB,C;K;IAGX,+B;MxBniDI,IAAI,EwB6iDI,KAAK,CxB7iDT,CAAJ,C;QACI,cwB4iDc,sD;QxB3iDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB4iDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAApB,C;K;IAGX,+B;MxBjjDI,IAAI,EwB2jDI,KAAK,CxB3jDT,CAAJ,C;QACI,cwB0jDc,sD;QxBzjDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB0jDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAApB,C;K;IAGX,+B;MxB/jDI,IAAI,EwBykDI,KAAK,CxBzkDT,CAAJ,C;QACI,cwBwkDc,sD;QxBvkDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBwkDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAApB,C;K;IAGX,mC;MxB7kDI,IAAI,EwBulDI,KAAK,CxBvlDT,CAAJ,C;QACI,cwBslDc,sD;QxBrlDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBslDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB3lDI,IAAI,EwBqmDI,KAAK,CxBrmDT,CAAJ,C;QACI,cwBomDc,sD;QxBnmDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBomDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBzmDI,IAAI,EwBmnDI,KAAK,CxBnnDT,CAAJ,C;QACI,cwBknDc,sD;QxBjnDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBknDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBvnDI,IAAI,EwBioDI,KAAK,CxBjoDT,CAAJ,C;QACI,cwBgoDc,sD;QxB/nDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBgoDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBroDI,IAAI,EwB+oDI,KAAK,CxB/oDT,CAAJ,C;QACI,cwB8oDc,sD;QxB7oDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB8oDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBnpDI,IAAI,EwB6pDI,KAAK,CxB7pDT,CAAJ,C;QACI,cwB4pDc,sD;QxB3pDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB4pDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBjqDI,IAAI,EwB2qDI,KAAK,CxB3qDT,CAAJ,C;QACI,cwB0qDc,sD;QxBzqDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB0qDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/qDI,IAAI,EwBysDI,KAAK,CxBzsDT,CAAJ,C;QACI,cwBwsDc,sD;QxBvsDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBwsDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB7sDI,IAAI,EwBuuDI,KAAK,CxBvuDT,CAAJ,C;QACI,cwBsuDc,sD;QxBruDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBsuDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB3uDI,IAAI,EwBqwDI,KAAK,CxBrwDT,CAAJ,C;QACI,cwBowDc,sD;QxBnwDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBowDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBzwDI,IAAI,EwBmyDI,KAAK,CxBnyDT,CAAJ,C;QACI,cwBkyDc,sD;QxBjyDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBkyDV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBvyDI,IAAI,EwBi0DI,KAAK,CxBj0DT,CAAJ,C;QACI,cwBg0Dc,sD;QxB/zDd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBg0DV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBr0DI,IAAI,EwB+1DI,KAAK,CxB/1DT,CAAJ,C;QACI,cwB81Dc,sD;QxB71Dd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB81DV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBn2DI,IAAI,EwB63DI,KAAK,CxB73DT,CAAJ,C;QACI,cwB43Dc,sD;QxB33Dd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB43DV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBj4DI,IAAI,EwB25DI,KAAK,CxB35DT,CAAJ,C;QACI,cwB05Dc,sD;QxBz5Dd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB05DV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/5DI,IAAI,EwBy7DI,KAAK,CxBz7DT,CAAJ,C;QACI,cwBw7Dc,sD;QxBv7Dd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBw7DV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB77DI,IAAI,EwBu9DI,KAAK,CxBv9DT,CAAJ,C;QACI,cwBs9Dc,sD;QxBr9Dd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBs9DV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB39DI,IAAI,EwBq+DI,KAAK,CxBr+DT,CAAJ,C;QACI,cwBo+Dc,sD;QxBn+Dd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBo+DV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBz+DI,IAAI,EwBm/DI,KAAK,CxBn/DT,CAAJ,C;QACI,cwBk/Dc,sD;QxBj/Dd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBk/DV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBv/DI,IAAI,EwBigEI,KAAK,CxBjgET,CAAJ,C;QACI,cwBggEc,sD;QxB//Dd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBggEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBrgEI,IAAI,EwB+hEI,KAAK,CxB/hET,CAAJ,C;QACI,cwB8hEc,sD;QxB7hEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB8hEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBniEI,IAAI,EwB6jEI,KAAK,CxB7jET,CAAJ,C;QACI,cwB4jEc,sD;QxB3jEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB4jEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBjkEI,IAAI,EwB2lEI,KAAK,CxB3lET,CAAJ,C;QACI,cwB0lEc,sD;QxBzlEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB0lEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/lEI,IAAI,EwBynEI,KAAK,CxBznET,CAAJ,C;QACI,cwBwnEc,sD;QxBvnEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBwnEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB7nEI,IAAI,EwBupEI,KAAK,CxBvpET,CAAJ,C;QACI,cwBspEc,sD;QxBrpEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBspEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB3pEI,IAAI,EwBqrEI,KAAK,CxBrrET,CAAJ,C;QACI,cwBorEc,sD;QxBnrEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBorEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBzrEI,IAAI,EwBmtEI,KAAK,CxBntET,CAAJ,C;QACI,cwBktEc,sD;QxBjtEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBktEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBvtEI,IAAI,EwBivEI,KAAK,CxBjvET,CAAJ,C;QACI,cwBgvEc,sD;QxB/uEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBgvEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBrvEI,IAAI,EwB+wEI,KAAK,CxB/wET,CAAJ,C;QACI,cwB8wEc,sD;QxB7wEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB8wEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBnxEI,IAAI,EwB6yEI,KAAK,CxB7yET,CAAJ,C;QACI,cwB4yEc,sD;QxB3yEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB4yEV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBjzEI,IAAI,EwB20EI,KAAK,CxB30ET,CAAJ,C;QACI,cwB00Ec,sD;QxBz0Ed,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB00EV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/0EI,IAAI,EwBy2EI,KAAK,CxBz2ET,CAAJ,C;QACI,cwBw2Ec,sD;QxBv2Ed,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBw2EV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB72EI,IAAI,EwBu4EI,KAAK,CxBv4ET,CAAJ,C;QACI,cwBs4Ec,sD;QxBr4Ed,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBs4EV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB34EI,IAAI,EwBq6EI,KAAK,CxBr6ET,CAAJ,C;QACI,cwBo6Ec,sD;QxBn6Ed,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBo6EV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBz6EI,IAAI,EwBm8EI,KAAK,CxBn8ET,CAAJ,C;QACI,cwBk8Ec,sD;QxBj8Ed,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBk8EV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBv8EI,IAAI,EwBi+EI,KAAK,CxBj+ET,CAAJ,C;QACI,cwBg+Ec,sD;QxB/9Ed,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBg+EV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBr+EI,IAAI,EwB+/EI,KAAK,CxB//ET,CAAJ,C;QACI,cwB8/Ec,sD;QxB7/Ed,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB8/EV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBngFI,IAAI,EwB6hFI,KAAK,CxB7hFT,CAAJ,C;QACI,cwB4hFc,sD;QxB3hFd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB4hFV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBjiFI,IAAI,EwB2jFI,KAAK,CxB3jFT,CAAJ,C;QACI,cwB0jFc,sD;QxBzjFd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB0jFV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/jFI,IAAI,EwBymFI,KAAK,CxBzmFT,CAAJ,C;QACI,cwBwmFc,sD;QxBvmFd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBwmFV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB7mFI,IAAI,EwBuoFI,KAAK,CxBvoFT,CAAJ,C;QACI,cwBsoFc,sD;QxBroFd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBsoFV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB3oFI,IAAI,EwBqqFI,KAAK,CxBrqFT,CAAJ,C;QACI,cwBoqFc,sD;QxBnqFd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBoqFV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBzqFI,IAAI,EwBmsFI,KAAK,CxBnsFT,CAAJ,C;QACI,cwBksFc,sD;QxBjsFd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBksFV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBvsFI,IAAI,EwBiuFI,KAAK,CxBjuFT,CAAJ,C;QACI,cwBguFc,sD;QxB/tFd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBguFV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBruFI,IAAI,EwB+vFI,KAAK,CxB/vFT,CAAJ,C;QACI,cwB8vFc,sD;QxB7vFd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB8vFV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBnwFI,IAAI,EwB6xFI,KAAK,CxB7xFT,CAAJ,C;QACI,cwB4xFc,sD;QxB3xFd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB4xFV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBjyFI,IAAI,EwB2zFI,KAAK,CxB3zFT,CAAJ,C;QACI,cwB0zFc,sD;QxBzzFd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB0zFV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/zFI,IAAI,EwBy1FI,KAAK,CxBz1FT,CAAJ,C;QACI,cwBw1Fc,sD;QxBv1Fd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBw1FV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB71FI,IAAI,EwBu3FI,KAAK,CxBv3FT,CAAJ,C;QACI,cwBs3Fc,sD;QxBr3Fd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBs3FV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB33FI,IAAI,EwBq5FI,KAAK,CxBr5FT,CAAJ,C;QACI,cwBo5Fc,sD;QxBn5Fd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBo5FV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBz5FI,IAAI,EwBm7FI,KAAK,CxBn7FT,CAAJ,C;QACI,cwBk7Fc,sD;QxBj7Fd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBk7FV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBv7FI,IAAI,EwBi9FI,KAAK,CxBj9FT,CAAJ,C;QACI,cwBg9Fc,sD;QxB/8Fd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBg9FV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBr9FI,IAAI,EwB++FI,KAAK,CxB/+FT,CAAJ,C;QACI,cwB8+Fc,sD;QxB7+Fd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB8+FV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBn/FI,IAAI,EwB6gGI,KAAK,CxB7gGT,CAAJ,C;QACI,cwB4gGc,sD;QxB3gGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB4gGV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBjhGI,IAAI,EwB2iGI,KAAK,CxB3iGT,CAAJ,C;QACI,cwB0iGc,sD;QxBziGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB0iGV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/iGI,IAAI,EwBykGI,KAAK,CxBzkGT,CAAJ,C;QACI,cwBwkGc,sD;QxBvkGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBwkGV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB7kGI,IAAI,EwBumGI,KAAK,CxBvmGT,CAAJ,C;QACI,cwBsmGc,sD;QxBrmGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBsmGV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB3mGI,IAAI,EwBqoGI,KAAK,CxBroGT,CAAJ,C;QACI,cwBooGc,sD;QxBnoGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBooGV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBzoGI,IAAI,EwBmqGI,KAAK,CxBnqGT,CAAJ,C;QACI,cwBkqGc,sD;QxBjqGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBkqGV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBvqGI,IAAI,EwBisGI,KAAK,CxBjsGT,CAAJ,C;QACI,cwBgsGc,sD;QxB/rGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBgsGV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBrsGI,IAAI,EwBuuGI,KAAK,CxBvuGT,CAAJ,C;QACI,cwBsuGc,sD;QxBruGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBsuGV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB3uGI,IAAI,EwBqwGI,KAAK,CxBrwGT,CAAJ,C;QACI,cwBowGc,sD;QxBnwGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBowGV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBzwGI,IAAI,EwBmyGI,KAAK,CxBnyGT,CAAJ,C;QACI,cwBkyGc,sD;QxBjyGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBkyGV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBvyGI,IAAI,EwBi0GI,KAAK,CxBj0GT,CAAJ,C;QACI,cwBg0Gc,sD;QxB/zGd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBg0GV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBr0GI,IAAI,EwB+1GI,KAAK,CxB/1GT,CAAJ,C;QACI,cwB81Gc,sD;QxB71Gd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB81GV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBn2GI,IAAI,EwB63GI,KAAK,CxB73GT,CAAJ,C;QACI,cwB43Gc,sD;QxB33Gd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB43GV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBj4GI,IAAI,EwB25GI,KAAK,CxB35GT,CAAJ,C;QACI,cwB05Gc,sD;QxBz5Gd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB05GV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/5GI,IAAI,EwBy7GI,KAAK,CxBz7GT,CAAJ,C;QACI,cwBw7Gc,sD;QxBv7Gd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBw7GV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB77GI,IAAI,EwBu9GI,KAAK,CxBv9GT,CAAJ,C;QACI,cwBs9Gc,sD;QxBr9Gd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBs9GV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB39GI,IAAI,EwBq+GI,KAAK,CxBr+GT,CAAJ,C;QACI,cwBo+Gc,sD;QxBn+Gd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBo+GV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBz+GI,IAAI,EwBm/GI,KAAK,CxBn/GT,CAAJ,C;QACI,cwBk/Gc,sD;QxBj/Gd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBk/GV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBv/GI,IAAI,EwBigHI,KAAK,CxBjgHT,CAAJ,C;QACI,cwBggHc,sD;QxB//Gd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBggHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBrgHI,IAAI,EwB+hHI,KAAK,CxB/hHT,CAAJ,C;QACI,cwB8hHc,sD;QxB7hHd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB8hHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBniHI,IAAI,EwB6jHI,KAAK,CxB7jHT,CAAJ,C;QACI,cwB4jHc,sD;QxB3jHd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB4jHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBjkHI,IAAI,EwB2lHI,KAAK,CxB3lHT,CAAJ,C;QACI,cwB0lHc,sD;QxBzlHd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB0lHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/lHI,IAAI,EwBynHI,KAAK,CxBznHT,CAAJ,C;QACI,cwBwnHc,sD;QxBvnHd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBwnHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB7nHI,IAAI,EwBupHI,KAAK,CxBvpHT,CAAJ,C;QACI,cwBspHc,sD;QxBrpHd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBspHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB3pHI,IAAI,EwBqrHI,KAAK,CxBrrHT,CAAJ,C;QACI,cwBorHc,sD;QxBnrHd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBorHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBzrHI,IAAI,EwBmtHI,KAAK,CxBntHT,CAAJ,C;QACI,cwBktHc,sD;QxBjtHd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBktHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBvtHI,IAAI,EwBivHI,KAAK,CxBjvHT,CAAJ,C;QACI,cwBgvHc,sD;QxB/uHd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBgvHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBrvHI,IAAI,EwB+wHI,KAAK,CxB/wHT,CAAJ,C;QACI,cwB8wHc,sD;QxB7wHd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB8wHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBnxHI,IAAI,EwB6yHI,KAAK,CxB7yHT,CAAJ,C;QACI,cwB4yHc,sD;QxB3yHd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB4yHV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBjzHI,IAAI,EwB20HI,KAAK,CxB30HT,CAAJ,C;QACI,cwB00Hc,sD;QxBz0Hd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB00HV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/0HI,IAAI,EwBy2HI,KAAK,CxBz2HT,CAAJ,C;QACI,cwBw2Hc,sD;QxBv2Hd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBw2HV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB72HI,IAAI,EwBu4HI,KAAK,CxBv4HT,CAAJ,C;QACI,cwBs4Hc,sD;QxBr4Hd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBs4HV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB34HI,IAAI,EwBq6HI,KAAK,CxBr6HT,CAAJ,C;QACI,cwBo6Hc,sD;QxBn6Hd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBo6HV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBz6HI,IAAI,EwBm8HI,KAAK,CxBn8HT,CAAJ,C;QACI,cwBk8Hc,sD;QxBj8Hd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBk8HV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBv8HI,IAAI,EwBi+HI,KAAK,CxBj+HT,CAAJ,C;QACI,cwBg+Hc,sD;QxB/9Hd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBg+HV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBr+HI,IAAI,EwB+/HI,KAAK,CxB//HT,CAAJ,C;QACI,cwB8/Hc,sD;QxB7/Hd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB8/HV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBnhII,IAAI,EwB6iII,KAAK,CxB7iIT,CAAJ,C;QACI,cwB4iIc,sD;QxB3iId,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB4iIV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBjjII,IAAI,EwB2kII,KAAK,CxB3kIT,CAAJ,C;QACI,cwB0kIc,sD;QxBzkId,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwB0kIV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB/kII,IAAI,EwBymII,KAAK,CxBzmIT,CAAJ,C;QACI,cwBwmIc,sD;QxBvmId,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBwmIV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB7mII,IAAI,EwBuoII,KAAK,CxBvoIT,CAAJ,C;QACI,cwBsoIc,sD;QxBroId,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBsoIV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxB3oII,IAAI,EwBqqII,KAAK,CxBrqIT,CAAJ,C;QACI,cwBoqIc,sD;QxBnqId,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBoqIV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBzqII,IAAI,EwBmsII,KAAK,CxBnsIT,CAAJ,C;QACI,cwBksIc,sD;QxBjsId,MAAM,gCAAyB,OAAQ,WAAjC,C;;MwBksIV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAc,CAAd,CAAhB,C;K;IAGX,mC;MxBvsII,IAAI,EwBiuII,KAAK,CxBjuIT,CAAJ,C;QACI,cwBguIc,sD;QxB/tId,MAAM,gCAAyB,OAAQ,WAAjC,C;;M

EX,C;mGAcA,yB;MAAA,+D;MAAA,uC;QAWW,kBAAgB,gB;QAi5HV,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UAl1HT,IA/DuC,SA+DnC,EAk1HkB,cAl1HlB,EAk1HkB,sBAl1HlB,WAk1H2B,IAl1H3B,CAAJ,C;YAA2C,sBAk1HZ,IAl1HY,C;;QA/D/C,OAiEO,W;O;KA5EX,C;mGAcA,yB;MAAA,+D;MAAA,uC;QAWW,kBAAgB,gB;QAg5HV,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UA90HT,IAlEwC,SAkEpC,EA80HkB,cA90HlB,EA80HkB,sBA90HlB,WA80H2B,IA90H3B,CAAJ,C;YAA2C,sBA80HZ,IA90HY,C;;QAlE/C,OAoEO,W;O;KA/EX,C;uGAcA,6C;MAs2HiB,gB;MADb,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QA11HT,IAAI,WA01HkB,cA11HlB,EA01HkB,sBA11HlB,WA01H2B,IA11H3B,CAAJ,C;UAA2C,sBA01HZ,IA11HY,C;;MAE/C,OAAO,W;K;uGAGX,6C;MAk2HiB,gB;MADb,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QA51HT,IAAI,WA41HkB,cA51HlB,EA41HkB,sBA51HlB,WA41H2B,IA51H3B,CAAJ,C;UAA2C,sBA41HZ,IA51HY,C;;MAE/C,OAAO,W;K;uGAGX,6C;MA81HiB,gB;MADb,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QAx1HT,IAAI,WAw1HkB,cAx1HlB,EAw1HkB,sBAx1HlB,WAw1H2B,IAx1H3B,CAAJ,C;UAA2C,sBAw1HZ,IAx1HY,C;;MAE/C,OAAO,W;K;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAgDH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAhD,YAAgD,SAAhD,gBAAgD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAhD,iD,OAAgD,iBAAhD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAiDH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAjD,YAAiD,SAAjD,gBAAiD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAjD,iD,OAAiD,iBAAjD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAkDH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAmDH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAoDH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAqDH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAsDH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAuDH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAwDH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAyDH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA0DH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA2DH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA4DH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA6DH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA8DH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA+DH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAgEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAiEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAkEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAmEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAoEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAqEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAsEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAuEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAwEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAyEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA0EH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA2EH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA4EH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA6EH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA8EH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA+EH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAgFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAiFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAkFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAmFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAoFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAqFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAsFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAuFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAwFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAyFH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA0FH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA2FH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA4FH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA6FH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA8FH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA+FH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAgGH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAiGH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAkGH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAmGH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAoGH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAqGH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAsGH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAuGH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAwGH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAyGH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA0GH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA2GH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA4GH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA6GH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA8GH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA+GH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAgHH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAiHH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAkHH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAmHH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAoHH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAqHH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAsHH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAuHH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAwHH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD,SAAlD,gBAAkD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAlD,iD,OAAkD,iBAAlD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QAyHH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAnD,YAAmD,SAAnD,gBAAmD,CAAV,U,OAAU,OAAV,CAAU,C;YAAA,yB,WAAA,WAAA,OAAA,C;;QAAnD,iD,OAAmD,iBAAnD,W;O;KAAA,C;2FAGX,yB;MAAA,+D;MAAA,uC;QASW,kBAAY,gB;QA0HH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAlD,YAAkD

AAiB,IAAjB,C;MACG,yB;MAAd,OAAc,cAAd,C;QAAc,uB;QACV,IAAK,WAAI,sBAAI,KAAJ,CAAJ,C;;MAET,OAAO,I;K;IAGX,sC;MASkB,Q;MAHd,WAAmB,wBAAR,OAAQ,EAAwB,EAAxB,B;MACnB,IAAI,SAAQ,CAAZ,C;QAAe,OAAO,W;MACtB,WAAW,iBAAkB,IAAlB,B;MACG,yB;MAAd,OAAc,cAAd,C;QAAc,uB;QACV,IAAK,WAAI,sBAAI,KAAJ,CAAJ,C;;MAET,OAAO,I;K;IAGX,2C;MAMI,OAAO,cAAkB,aAAR,iBAAQ,EAAW,OAAX,CAAlB,B;MAAmC,C;K;IAGxC,2C;MAMI,OAAO,eAAmB,aAAR,iBAAQ,EAAW,OAAX,CAAnB,B;MAAoC,C;K;IAGzC,2C;MAMI,OAAO,eAAmB,aAAR,iBAAQ,EAAW,OAAX,CAAnB,B;MAAoC,C;K;IAGzC,2C;MAMI,OAAO,gBAAoB,aAAR,iBAAQ,EAAW,OAAX,CAApB,B;MAAqC,C;K;IAG1C,2C;MAMI,OAAO,cAAkB,aAAR,iBAAQ,EAAW,OAAX,CAAlB,B;MAAmC,C;K;IAGxC,2C;MAMI,OAAO,eAAmB,aAAR,iBAAQ,EAAW,OAAX,CAAnB,B;MAAoC,C;K;IAGzC,2C;MAMI,OAAO,eAAmB,aAAR,iBAAQ,EAAW,OAAX,CAAnB,B;MAAoC,C;K;IAGzC,2C;MAMI,OAAO,gBAAoB,aAAR,iBAAQ,EAAW,OAAX,CAApB,B;MAAqC,C;K;IAG1C,+B;MAgBiB,Q;MxB7xEb,IAAI,EwBuxEI,KAAK,CxBvxET,CAAJ,C;QACI,cwBsxEc,sD;QxBrxEd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBsxEV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBvyEb,IAAI,EwBiyEI,KAAK,CxBjyET,CAAJ,C;QACI,cwBgyEc,sD;QxB/xEd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBgyEV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBjzEb,IAAI,EwB2yEI,KAAK,CxB3yET,CAAJ,C;QACI,cwB0yEc,sD;QxBzyEd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB0yEV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB30Eb,IAAI,EwBq0EI,KAAK,CxBr0ET,CAAJ,C;QACI,cwBo0Ec,sD;QxBn0Ed,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBo0EV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB71Eb,IAAI,EwBu1EI,KAAK,CxBv1ET,CAAJ,C;QACI,cwBs1Ec,sD;QxBr1Ed,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBs1EV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB/2Eb,IAAI,EwBy2EI,KAAK,CxBz2ET,CAAJ,C;QACI,cwBw2Ec,sD;QxBv2Ed,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBw2EV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBj4Eb,IAAI,EwB23EI,KAAK,CxB33ET,CAAJ,C;QACI,cwB03Ec,sD;QxBz3Ed,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB03EV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBn5Eb,IAAI,EwB65EI,KAAK,CxB75ET,CAAJ,C;QACI,cwB45Ec,sD;QxB35Ed,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB45EV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBr7Eb,IAAI,EwB+6EI,KAAK,CxB/6ET,CAAJ,C;QACI,cwB86Ec,sD;QxB76Ed,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB86EV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBv8Eb,IAAI,EwBi8EI,KAAK,CxBj8ET,CAAJ,C;QACI,cwBg8Ec,sD;QxB/7Ed,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBg8EV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBz9Eb,IAAI,EwBm9EI,KAAK,CxBn9ET,CAAJ,C;QACI,cwBk9Ec,sD;QxBj9Ed,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBk9EV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB3+Eb,IAAI,EwBq+EI,KAAK,CxBr+ET,CAAJ,C;QACI,cwBo+Ec,sD;QxBn+Ed,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBo+EV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB7/Eb,IAAI,EwBu/EI,KAAK,CxBv/ET,CAAJ,C;QACI,cwBs/Ec,sD;QxBr/Ed,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBs/EV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBjhFb,IAAI,EwB2gFI,KAAK,CxB3gFT,CAAJ,C;QACI,cwB0gFc,sD;QxBzgFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB0gFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBrjFb,IAAI,EwB+iFI,KAAK,CxB/iFT,CAAJ,C;QACI,cwB8iFc,sD;QxB7iFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB8iFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBzkFb,IAAI,EwBmkFI,KAAK,CxBnkFT,CAAJ,C;QACI,cwBkkFc,sD;QxBjkFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBkkFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB7lFb,IAAI,EwBulFI,KAAK,CxBvlFT,CAAJ,C;QACI,cwBslFc,sD;QxBrlFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBslFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBjnFb,IAAI,EwB2mFI,KAAK,CxB3mFT,CAAJ,C;QACI,cwB0mFc,sD;QxBzmFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB0mFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBrpFb,IAAI,EwB+oFI,KAAK,CxB/oFT,CAAJ,C;QACI,cwB8oFc,sD;QxB7oFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB8oFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBzqFb,IAAI,EwBmqFI,KAAK,CxBnqFT,CAAJ,C;QACI,cwBkqFc,sD;QxBjqFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBkqFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB7rFb,IAAI,EwBurFI,KAAK,CxBvrFT,CAAJ,C;QACI,cwBsrFc,sD;QxBrrFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBsrFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBjtFb,IAAI,EwB2sFI,KAAK,CxB3sFT,CAAJ,C;QACI,cwB0sFc,sD;QxBzsFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB0sFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBruFb,IAAI,EwB+tFI,KAAK,CxB/tFT,CAAJ,C;QACI,cwB8tFc,sD;QxB7tFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB8tFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBzvFb,IAAI,EwBmvFI,KAAK,CxBnvFT,CAAJ,C;QACI,cwBkvFc,sD;QxBjvFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBkvFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB7wFb,IAAI,EwBuwFI,KAAK,CxBvwFT,CAAJ,C;QACI,cwBswFc,sD;QxBrwFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBswFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBjyFb,IAAI,EwB2xFI,KAAK,CxB3xFT,CAAJ,C;QACI,cwB0xFc,sD;QxBzxFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB0xFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBrzFb,IAAI,EwB+yFI,KAAK,CxB/yFT,CAAJ,C;QACI,cwB8yFc,sD;QxB7yFd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB8yFV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBz0Fb,IAAI,EwBm0FI,KAAK,CxBn0FT,CAAJ,C;QACI,cwBk0Fc,sD;QxBj0Fd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBk0FV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB71Fb,IAAI,EwBu1FI,KAAK,CxBv1FT,CAAJ,C;QACI,cwBs1Fc,sD;QxBr1Fd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBs1FV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBj3Fb,IAAI,EwB22FI,KAAK,CxB32FT,CAAJ,C;QACI,cwB02Fc,sD;QxBz2Fd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB02FV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBn4Fb,IAAI,EwB63FI,KAAK,CxB73FT,CAAJ,C;QACI,cwB43Fc,sD;QxB33Fd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB43FV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBr5Fb,IAAI,EwB+4FI,KAAK,CxB/4FT,CAAJ,C;QACI,cwB84Fc,sD;QxB74Fd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB84FV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBv6Fb,IAAI,EwBi6FI,KAAK,CxBj6FT,CAAJ,C;QACI,cwBg6Fc,sD;QxB/5Fd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBg6FV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBz7Fb,IAAI,EwBm7FI,KAAK,CxBn7FT,CAAJ,C;QACI,cwBk7Fc,sD;QxBj7Fd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBk7FV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB39Fb,IAAI,EwBq9FI,KAAK,CxBr9FT,CAAJ,C;QACI,cwBo9Fc,sD;QxBn9Fd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBo9FV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB7+Fb,IAAI,EwBu+FI,KAAK,CxBv+FT,CAAJ,C;QACI,cwBs+Fc,sD;QxBr+Fd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBs+FV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBjgGb,IAAI,EwB2/FI,KAAK,CxB3/FT,CAAJ,C;QACI,cwB0/Fc,sD;QxBz/Fd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB0/FV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBrhGb,IAAI,EwB+gGI,KAAK,CxB/gGT,CAAJ,C;QACI,cwB8gGc,sD;QxB7gGd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB8gGV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBziGb,IAAI,EwBmiGI,KAAK,CxBniGT,CAAJ,C;QACI,cwBkiGc,sD;QxBjiGd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBkiGV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB7jGb,IAAI,EwBujGI,KAAK,CxBvjGT,CAAJ,C;QACI,cwBsjGc,sD;QxBrjGd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBsjGV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBrlGb,IAAI,EwB+kGI,KAAK,CxB/kGT,CAAJ,C;QACI,cwB8kGc,sD;QxB7kGd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwB8kGV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxBzmGb,IAAI,EwBmmGI,KAAK,CxBnmGT,CAAJ,C;QACI,cwBkmGc,sD;QxBjmGd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBkmGV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;QAAY,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,B;MACX,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAK,WAAI,IAAJ,C;QAAS,OAAO,mCAAf,C;MAAoB,C;K;IAGzB,+B;MAgBiB,Q;MxB7nGb,IAAI,EwBunGI,KAAK,CxBvnGT,CAAJ,C;QACI,cwBsnGc,sD;QxBrnGd,MAAM,gBAAyB,OAAQ,WAAjC,C;;MwBsnGV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,CAAT,C;Q

SI,iBAo1CgB,cAAR,iBAAQ,CAp1ChB,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,iB;O;KAdX,C;mGAiBA,yB;MAAA,4C;MAAA,gD;MAo1CI,8D;MAp1CJ,uC;QASI,iBA20CgB,cAAR,iBAAQ,CA30ChB,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,iB;O;KAdX,C;mGAiBA,yB;MAAA,4C;MAAA,gD;MA20CI,8D;MA30CJ,uC;QASI,iBAk0CgB,cAAR,iBAAQ,CAl0ChB,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,iB;O;KAdX,C;2FAiBA,yB;MAAA,+D;MAAA,uC;QAUI,iBAAO,cAAP,iBAAO,CAAP,WAAO,CAAP,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,iB;O;KAdX,C;2FAiBA,yB;MAAA,+D;MAAA,uC;QAUI,iBAAO,cAAP,iBAAO,CAAP,WAAO,CAAP,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,iB;O;KAdX,C;uFAiBA,yB;MAAA,2

AO,S;MACD,kBA4lBd,eAAmB,UA5lBnB,SA4lBW,QAAQ,CAAnB,C;MA5lBsB,mB;MAA7B,OrBt6FO,W;K;IqBy6FX,kC;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8lBd,eA9lBA,SA8lBW,QvBx/EM,QuBw/EjB,C;MA9lBsB,mB;MAA7B,OrBh7FO,W;K;IqBm7FX,mC;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgmBd,gBAAhmBA,SAAgmBY,QvB1/EK,QuB0/EjB,C;MAhmBsB,mBAAmB,MAA7B,OrB17FO,W;K;IqB67FX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkmBd,kBAAlmBA,SAAkmBU,QvBpgFO,QuBogFjB,C;MAlmBsB,mBAAmB,MAA7B,OrBp8FO,W;K;IqBu8FX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAomBd,kBAApmBA,SAomBU,QvB9gFO,QuB8gFjB,C;MApmBsB,mBAAmB,MAA7B,OrB98FO,W;K;IqBi9FX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAsmBd,kBAtmBA,SAsmBU,QvBxhFO,QuBwhFjB,C;MAtmBsB,mBAAmB,MAA7B,OrBx9FO,W;K;IqB29FX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwmBd,kBAxmBA,SAwmBU,QvBliFO,QuBkiFjB,C;MAxmBsB,mBAAmB,MAA7B,OrBl+FO,W;K;IqBq+FX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0mBd,kBA1mBA,SA0mBU,QvB5iFO,QuB4iFjB,C;MA1mBsB,mBAAmB,MAA7B,OrB5+FO,W;K;IqB++FX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4mBd,kBA5mBA,SA4mBU,QvBtjFO,QuBsjFjB,C;MA5mBsB,mBAAmB,MAA7B,OrBt/FO,W;K;IqBy/FX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8mBd,kBA9mBA,SA8mBU,QvBhkFO,QuBgkFjB,C;MA9mBsB,mBAAmB,MAA7B,OrBhgGO,W;K;IqBmgGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgnBd,kBAhnBA,SAgnBU,QvB1kFO,QuB0kFjB,C;MAhnBsB,mBAAmB,MAA7B,OrB1gGO,W;K;IqB6gGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAknBd,kBAlnBA,SAknBU,QvB5lFO,QuB4lFjB,C;MAlnBsB,mBAAmB,MAA7B,OrB5hGO,W;K;IqB+hGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAonBd,kBApnBA,SAonBU,QvB9mFO,QuB8mFjB,C;MApnBsB,mBAAmB,MAA7B,OrB9iGO,W;K;IqBijGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAsnBd,kBAtnBA,SAsnBU,QvBxnFO,QuBwnFjB,C;MAtnBsB,mBAAmB,MAA7B,OrBxjGO,W;K;IqB2jGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwnBd,kBAxnBA,SAwnBU,QvBloFO,QuBkoFjB,C;MAxnBsB,mBAAmB,MAA7B,OrBlkGO,W;K;IqBqkGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0nBd,kBA1nBA,SA0nBU,QvB5oFO,QuB4oFjB,C;MA1nBsB,mBAAmB,MAA7B,OrB5kGO,W;K;IqB+kGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4nBd,kBA5nBA,SA4nBU,QvBtpFO,QuBspFjB,C;MA5nBsB,mBAAmB,MAA7B,OrBtlGO,W;K;IqBylGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8nBd,kBA9nBA,SA8nBU,QvBhqFO,QuBgqFjB,C;MA9nBsB,mBAAmB,MAA7B,OrBhmGO,W;K;IqBmmGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgoBd,kBAhoBA,SAgoBU,QvB1qFO,QuB0qFjB,C;MAhoBsB,mBAAmB,MAA7B,OrB1mGO,W;K;IqB6mGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkoBd,kBAloBA,SAkoBU,QvB5rFO,QuB4rFjB,C;MAloBsB,mBAAmB,MAA7B,OrB5nGO,W;K;IqB+nGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAooBd,kBApoBA,SAooBU,QvB9sFO,QuB8sFjB,C;MApoBsB,mBAAmB,MAA7B,OrB9oGO,W;K;IqBipGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAsoBd,kBAtoBA,SAsoBU,QvBxtFO,QuBwtFjB,C;MAtoBsB,mBAAmB,MAA7B,OrBxpGO,W;K;IqB2pGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwoBd,kBAxoBA,SAwoBU,QvBluFO,QuBkuFjB,C;MAxoBsB,mBAAmB,MAA7B,OrBlqGO,W;K;IqBqqGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0oBd,kBA1oBA,SA0oBU,QvB5uFO,QuB4uFjB,C;MA1oBsB,mBAAmB,MAA7B,OrB5qGO,W;K;IqB+qGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4oBd,kBA5oBA,SA4oBU,QvBtvFO,QuBsvFjB,C;MA5oBsB,mBAAmB,MAA7B,OrBtrGO,W;K;IqBysGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8oBd,kBA9oBA,SA8oBU,QvBhwFO,QuBgwFjB,C;MA9oBsB,mBAAmB,MAA7B,OrBhsGO,W;K;IqBmsGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgpBd,kBAhpBA,SAgpBU,QvB1wFO,QuB0wFjB,C;MAhpBsB,mBAAmB,MAA7B,OrB1sGO,W;K;IqB6sGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkpBd,kBAlpBA,SAkpBU,QvB5xFO,QuB4xFjB,C;MAlpBsB,mBAAmB,MAA7B,OrB5tGO,W;K;IqB+tGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAopBd,kBAppBA,SAopBU,QvB9yFO,QuB8yFjB,C;MAppBsB,mBAAmB,MAA7B,OrB9uGO,W;K;IqBivGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAspBd,kBAtpBA,SAspBU,QvBxzFO,QuBwzFjB,C;MAtpBsB,mBAAmB,MAA7B,OrBxvGO,W;K;IqB2vGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwpBd,kBAxpBA,SAwpBU,QvBl0FO,QuBk0FjB,C;MAxpBsB,mBAAmB,MAA7B,OrBlwGO,W;K;IqBqwGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0pBd,kBA1pBA,SA0pBU,QvB50FO,QuB40FjB,C;MA1pBsB,mBAAmB,MAA7B,OrB5wGO,W;K;IqB+wGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4pBd,kBA5pBA,SA4pBU,QvBt1FO,QuBs1FjB,C;MA5pBsB,mBAAmB,MAA7B,OrBtxGO,W;K;IqBwxGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8pBd,kBA9pBA,SA8pBU,QvBh2FO,QuBg2FjB,C;MA9pBsB,mBAAmB,MAA7B,OrBhyGO,W;K;IqBmyGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgqBd,kBAhqBA,SAgqBU,QvB12FO,QuB02FjB,C;MAhqBsB,mBAAmB,MAA7B,OrB1yGO,W;K;IqB6yGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkqBd,kBAlqBA,SAkqBU,QvB53FO,QuB43FjB,C;MAlqBsB,mBAAmB,MAA7B,OrB5yGO,W;K;IqB+yGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAoqBd,kBApqBA,SAoqBU,QvB94FO,QuB84FjB,C;MApqBsB,mBAAmB,MAA7B,OrB9yGO,W;K;IqBizGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAsqBd,kBAtqBA,SAsqBU,QvBx5FO,QuBw5FjB,C;MAtqBsB,mBAAmB,MAA7B,OrBxzGO,W;K;IqB2zGX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwqBd,kBAxqBA,SAwqBU,QvBl6FO,QuBk6FjB,C;MAxqBsB,mBAAmB,MAA7B,OrBl0GO,W;K;IqBq0GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0qBd,kBA1qBA,SA0qBU,QvB56FO,QuB46FjB,C;MA1qBsB,mBAAmB,MAA7B,OrB50GO,W;K;IqB+0GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4qBd,kBA5qBA,SA4qBU,QvBt7FO,QuBs7FjB,C;MA5qBsB,mBAAmB,MAA7B,OrBt1GO,W;K;IqBy1GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8qBd,kBA9qBA,SA8qBU,QvBh8FO,QuBg8FjB,C;MA9qBsB,mBAAmB,MAA7B,OrBh2GO,W;K;IqBm2GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgrBd,kBAhrBA,SAgrBU,QvB18FO,QuB08FjB,C;MAhrBsB,mBAAmB,MAA7B,OrB12GO,W;K;IqB62GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkrBd,kBAlrBA,SAkrBU,QvB59FO,QuB49FjB,C;MAlrBsB,mBAAmB,MAA7B,OrB52GO,W;K;IqB+2GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAorBd,kBAprBA,SAorBU,QvB9+FO,QuB8+FjB,C;MAprBsB,mBAAmB,MAA7B,OrB93GO,W;K;IqBi4GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAsrBd,kBAtrBA,SAsrBU,QvBxgGO,QuBwgGjB,C;MAtrBsB,mBAAmB,MAA7B,OrBx4GO,W;K;IqB24GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwrBd,kBAxrBA,SAwrBU,QvBlhGO,QuBkhGjB,C;MAxrBsB,mBAAmB,MAA7B,OrBl5GO,W;K;IqBq5GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0rBd,kBA1rBA,SA0rBU,QvB5hGO,QuB4hGjB,C;MA1rBsB,mBAAmB,MAA7B,OrB55GO,W;K;IqB+5GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4rBd,kBA5rBA,SA4rBU,QvBtiGO,QuBsiGjB,C;MA5rBsB,mBAAmB,MAA7B,OrBt6GO,W;K;IqBy6GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8rBd,kBA9rBA,SA8rBU,QvBhjGO,QuBgjGjB,C;MA9rBsB,mBAAmB,MAA7B,OrBh7GO,W;K;IqBm7GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgsBd,kBAhsBA,SAgsBU,QvB1jGO,QuB0jGjB,C;MAhsBsB,mBAAmB,MAA7B,OrB17GO,W;K;IqB67GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAksBd,kBAlsBA,SAksBU,QvB5kGO,QuB4kGjB,C;MAlsBsB,mBAAmB,MAA7B,OrB58GO,W;K;IqB+8GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAosBd,kBApsBA,SAosBU,QvB9lGO,QuB8lGjB,C;MApsBsB,mBAAmB,MAA7B,OrB99GO,W;K;IqBi+GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAssBd,kBAtsBA,SAssBU,QvBxmGO,QuBwmGjB,C;MAtsBsB,mBAAmB,MAA7B,OrBx+GO,W;K;IqB2+GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwsBd,kBAxsBA,SAwsBU,QvBlnGO,QuBknGjB,C;MAxsBsB,mBAAmB,MAA7B,OrBl/GO,W;K;IqBq/GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0sBd,kBA1sBA,SA0sBU,QvB5nGO,QuB4nGjB,C;MA1sBsB,mBAAmB,MAA7B,OrB5/GO,W;K;IqB+/GX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4sBd,kBA5sBA,SA4sBU,QvBtoGO,QuBsoGjB,C;MA5sBsB,mBAAmB,MAA7B,OrBtgHO,W;K;IqBygHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8sBd,kBA9sBA,SA8sBU,QvBhpGO,QuBgpGjB,C;MA9sBsB,mBAAmB,MAA7B,OrBhhHO,W;K;IqBmhHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgtBd,kBAhtBA,SAgtBU,QvB1pGO,QuB0pGjB,C;MAhtBsB,mBAAmB,MAA7B,OrB1hHO,W;K;IqB6hHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAktBd,kBAltBA,SAktBU,QvB5qGO,QuB4qGjB,C;MAltBsB,mBAAmB,MAA7B,OrB5iHO,W;K;IqB+iHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAotBd,kBAptBA,SAotBU,QvB9rGO,QuB8rGjB,C;MAptBsB,mBAAmB,MAA7B,OrB9jHO,W;K;IqBikHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAstBd,kBAttBA,SAstBU,QvBxsGO,QuBwsGjB,C;MAttBsB,mBAAmB,MAA7B,OrBxkHO,W;K;IqB2kHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwtBd,kBAxtBA,SAwtBU,QvBltGO,QuBktGjB,C;MAxtBsB,mBAAmB,MAA7B,OrBllHO,W;K;IqBqlHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0tBd,kBA1tBA,SA0tBU,QvB5tGO,QuB4tGjB,C;MA1tBsB,mBAAmB,MAA7B,OrB5lHO,W;K;IqB+lHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4tBd,kBA5tBA,SA4tBU,QvBtuGO,QuBsuGjB,C;MA5tBsB,mBAAmB,MAA7B,OrBtmHO,W;K;IqBymHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8tBd,kBA9tBA,SA8tBU,QvBhvGO,QuBgvGjB,C;MA9tBsB,mBAAmB,MAA7B,OrBhnHO,W;K;IqBmnHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAguBd,kBAhuBA,SAguBU,QvB1vGO,QuB0vGjB,C;MAhuBsB,mBAAmB,MAA7B,OrB1nHO,W;K;IqB6nHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkuBd,kBAluBA,SAkuBU,QvB5wGO,QuB4wGjB,C;MAluBsB,mBAAmB,MAA7B,OrB5oHO,W;K;IqB+oHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAouBd,kBApuBA,SAouBU,QvB9xGO,QuB8xGjB,C;MApuBsB,mBAAmB,MAA7B,OrB9pHO,W;K;IqBiqHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAsuBd,kBAtuBA,SAsuBU,QvBxyGO,QuBwyGjB,C;MAtuBsB,mBAAmB,MAA7B,OrBxqHO,W;K;IqB2qHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwuBd,kBAxuBA,SAwuBU,QvBlzGO,QuBkzGjB,C;MAxuBsB,mBAAmB,MAA7B,OrBlrHO,W;K;IqBqrHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0uBd,kBA1uBA,SA0uBU,QvB50GO,QuB40GjB,C;MA1uBsB,mBAAmB,MAA7B,OrB5rHO,W;K;IqB+rHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4uBd,kBA5uBA,SA4uBU,QvBt1GO,QuBs1GjB,C;MA5uBsB,mBAAmB,MAA7B,OrBtsHO,W;K;IqBysHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8uBd,kBA9uBA,SA8uBU,QvBh2GO,QuBg2GjB,C;MA9uBsB,mBAAmB,MAA7B,OrBhtHO,W;K;IqBmtHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgvBd,kBAhvBA,SAgvBU,QvB12GO,QuB02GjB,C;MAhvBsB,mBAAmB,MAA7B,OrB1tHO,W;K;IqB6tHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkvBd,kBAlvBA,SAkvBU,QvB53GO,QuB43GjB,C;MAlvBsB,mBAAmB,MAA7B,OrB5tHO,W;K;IqB+tHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAovBd,kBApvBA,SAovBU,QvB94GO,QuB84GjB,C;MApvBsB,mBAAmB,MAA7B,OrB9tHO,W;K;IqBiuHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAsvBd,kBAtvBA,SAsvBU,QvBx5GO,QuBw5GjB,C;MAtvBsB,mBAAmB,MAA7B,OrBxuHO,W;K;IqB2uHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwvBd,kBAxvBA,SAwvBU,QvBl6GO,QuBk6GjB,C;MAxvBsB,mBAAmB,MAA7B,OrBlvHO,W;K;IqBqvHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0vBd,kBA1vBA,SA0vBU,QvB56GO,QuB46GjB,C;MA1vBsB,mBAAmB,MAA7B,OrB5vHO,W;K;IqB+vHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4vBd,kBA5vBA,SA4vBU,QvBt7GO,QuBs7GjB,C;MA5vBsB,mBAAmB,MAA7B,OrBtwHO,W;K;IqBywHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8vBd,kBA9vBA,SA8vBU,QvBh8GO,QuBg8GjB,C;MA9vBsB,mBAAmB,MAA7B,OrBhxHO,W;K;IqBmxHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgwBd,kBAhwBA,SAgwBU,QvB18GO,QuB08GjB,C;MAhwBsB,mBAAmB,MAA7B,OrB1xHO,W;K;IqB6xHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkwBd,kBAlwBA,SAkwBU,QvB59GO,QuB49GjB,C;MAlwBsB,mBAAmB,MAA7B,OrB5xHO,W;K;IqB+xHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAowBd,kBApwBA,SAowBU,QvB9+GO,QuB8+GjB,C;MApwBsB,mBAAmB,MAA7B,OrB9yHO,W;K;IqBizHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAswBd,kBAtwBA,SAswBU,QvBxgHO,QuBwgHjB,C;MAtwBsB,mBAAmB,MAA7B,OrBxzHO,W;K;IqB2zHX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwwBd,kBAxwBA,SAwwBU,QvBlhHO,QuBkhHjB,C;MAxwBsB,mBAAmB,MAA7B,OrBl0HO,W;K;IqBq0HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0wBd,kBA1wBA,SA0wBU,QvB5hHO,QuB4hHjB,C;MA1wBsB,mBAAmB,MAA7B,OrB50HO,W;K;IqB+0HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4wBd,kBA5wBA,SA4wBU,QvBtiHO,QuBsiHjB,C;MA5wBsB,mBAAmB,MAA7B,OrBt1HO,W;K;IqBy1HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8wBd,kBA9wBA,SA8wBU,QvBhjHO,QuBgjHjB,C;MA9wBsB,mBAAmB,MAA7B,OrBh2HO,W;K;IqBm2HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgxBd,kBAhxBA,SAgxBU,QvB1jHO,QuB0jHjB,C;MAhxBsB,mBAAmB,MAA7B,OrB12HO,W;K;IqB62HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkxBd,kBAlxBA,SAkxBU,QvB5kHO,QuB4kHjB,C;MAlxBsB,mBAAmB,MAA7B,OrB52HO,W;K;IqB+2HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAoxBd,kBApxBA,SAoxBU,QvB9lHO,QuB8lHjB,C;MApxBsB,mBAAmB,MAA7B,OrB92HO,W;K;IqBi3HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAsxBd,kBAtxBA,SAsxBU,QvBxmHO,QuBwmHjB,C;MAtxBsB,mBAAmB,MAA7B,OrBx3HO,W;K;IqB23HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwxBd,kBAxxBA,SAwxBU,QvBlnHO,QuBknHjB,C;MAxxBsB,mBAAmB,MAA7B,OrBl4HO,W;K;IqBq4HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0xBd,kBA1xBA,SA0xBU,QvB5nHO,QuB4nHjB,C;MA1xBsB,mBAAmB,MAA7B,OrB54HO,W;K;IqB+4HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4xBd,kBA5xBA,SA4xBU,QvBtoHO,QuBsoHjB,C;MA5xBsB,mBAAmB,MAA7B,OrBt5HO,W;K;IqBy5HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8xBd,kBA9xBA,SA8xBU,QvBhpHO,QuBgpHjB,C;MA9xBsB,mBAAmB,MAA7B,OrBh6HO,W;K;IqBm6HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgyBd,kBAhyBA,SAgyBU,QvB1pHO,QuB0pHjB,C;MAhyBsB,mBAAmB,MAA7B,OrB16HO,W;K;IqB66HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkyBd,kBAlyBA,SAkyBU,QvB5qHO,QuB4qHjB,C;MAlyBsB,mBAAmB,MAA7B,OrB56HO,W;K;IqB+6HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAoyBd,kBApyBA,SAoyBU,QvB9rHO,QuB8rHjB,C;MApyBsB,mBAAmB,MAA7B,OrB96HO,W;K;IqBi7HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAsyBd,kBAtyBA,SAsyBU,QvBxsHO,QuBwsHjB,C;MAtyBsB,mBAAmB,MAA7B,OrBx7HO,W;K;IqB27HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwyBd,kBAxyBA,SAwyBU,QvBltHO,QuBktHjB,C;MAxyBsB,mBAAmB,MAA7B,OrBl8HO,W;K;IqBq8HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0yBd,kBA1yBA,SA0yBU,QvB5tHO,QuB4tHjB,C;MA1yBsB,mBAAmB,MAA7B,OrB58HO,W;K;IqB+8HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4yBd,kBA5yBA,SA4yBU,QvBtuHO,QuBsuHjB,C;MA5yBsB,mBAAmB,MAA7B,OrBt9HO,W;K;IqBy9HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8yBd,kBA9yBA,SA8yBU,QvBhvHO,QuBgvHjB,C;MA9yBsB,mBAAmB,MAA7B,OrBh+HO,W;K;IqBm+HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAgzBd,kBAhzBA,SAgzBU,QvB1vHO,QuB0vHjB,C;MAhzBsB,mBAAmB,MAA7B,OrB1+HO,W;K;IqB6+HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAkzBd,kBAlzBA,SAkzBU,QvB5wHO,QuB4wHjB,C;MAlzBsB,mBAAmB,MAA7B,OrB5/HO,W;K;IqB+/HX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAozBd,kBApzBA,SAozBU,QvB9xHO,QuB8xHjB,C;MApzBsB,mBAAmB,MAA7B,OrB9gIO,W;K;IqBihIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAszBd,kBAtzBA,SAszBU,QvBxyHO,QuBwyHjB,C;MAtzBsB,mBAAmB,MAA7B,OrBxhIO,W;K;IqB2hIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAwzBd,kBAxzBA,SAwzBU,QvBlzHO,QuBkzHjB,C;MAxzBsB,mBAAmB,MAA7B,OrBliIO,W;K;IqBqiIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA0zBd,kBA1zBA,SA0zBU,QvB50HO,QuB40HjB,C;MA1zBsB,mBAAmB,MAA7B,OrB5iIO,W;K;IqB+iIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA4zBd,kBA5zBA,SA4zBU,QvBt1HO,QuBs1HjB,C;MA5zBsB,mBAAmB,MAA7B,OrBtjIO,W;K;IqByjIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA8zBd,kBA9zBA,SA8zBU,QvBh2HO,QuBg2HjB,C;MA9zBsB,mBAAmB,MAA7B,OrBhkIO,W;K;IqBmkIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAg0Bd,kBAh0BA,SAg0BU,QvB12HO,QuB02HjB,C;MAh0BsB,mBAAmB,MAA7B,OrB1kIO,W;K;IqB6kIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAk0Bd,kBAl0BA,SAk0BU,QvB53HO,QuB43HjB,C;MAl0BsB,mBAAmB,MAA7B,OrB5kIO,W;K;IqB+kIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAo0Bd,kBAp0BA,SAo0BU,QvB94HO,QuB84HjB,C;MAp0BsB,mBAAmB,MAA7B,OrB9kIO,W;K;IqBilIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAs0Bd,kBAt0BA,SAs0BU,QvBx5HO,QuBw5HjB,C;MAt0BsB,mBAAmB,MAA7B,OrBxlIO,W;K;IqB2lIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAw0Bd,kBAx0BA,SAw0BU,QvBl6HO,QuBk6HjB,C;MAx0BsB,mBAAmB,MAA7B,OrBlmIO,W;K;IqBqmIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA00Bd,kBA10BA,SA00BU,QvB56HO,QuB46HjB,C;MA10BsB,mBAAmB,MAA7B,OrB5mIO,W;K;IqB+mIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA40Bd,kBA50BA,SA40BU,QvBt7HO,QuBs7HjB,C;MA50BsB,mBAAmB,MAA7B,OrBtnIO,W;K;IqBynIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA80Bd,kBA90BA,SA80BU,QvBh8HO,QuBg8HjB,C;MA90BsB,mBAAmB,MAA7B,OrBhoIO,W;K;IqBmoIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAg1Bd,kBAh1BA,SAg1BU,QvB18HO,QuB08HjB,C;MAh1BsB,mBAAmB,MAA7B,OrB1oIO,W;K;IqB6oIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAk1Bd,kBAl1BA,SAk1BU,QvB59HO,QuB49HjB,C;MAl1BsB,mBAAmB,MAA7B,OrB5oIO,W;K;IqB+oIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAo1Bd,kBAp1BA,SAo1BU,QvB9+HO,QuB8+HjB,C;MAp1BsB,mBAAmB,MAA7B,OrB9oIO,W;K;IqBipIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAs1Bd,kBAt1BA,SAs1BU,QvBxgIO,QuBwgIjB,C;MAt1BsB,mBAAmB,MAA7B,OrBxpIO,W;K;IqB2pIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAw1Bd,kBAx1BA,SAw1BU,QvBlhIO,QuBkhIjB,C;MAx1BsB,mBAAmB,MAA7B,OrBlqIO,W;K;IqBqqIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA01Bd,kBA11BA,SA01BU,QvB5hIO,QuB4hIjB,C;MA11BsB,mBAAmB,MAA7B,OrB5qIO,W;K;IqB+qIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA41Bd,kBA51BA,SA41BU,QvBtiIO,QuBsiIjB,C;MA51BsB,mBAAmB,MAA7B,OrBtrIO,W;K;IqBwrIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBA81Bd,kBA91BA,SA81BU,QvBhjIO,QuBgjIjB,C;MA91BsB,mBAAmB,MAA7B,OrBhsIO,W;K;IqBmsIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAg2Bd,kBAh2BA,SAg2BU,QvB1jIO,QuB0jIjB,C;MAh2BsB,mBAAmB,MAA7B,OrB1sIO,W;K;IqB6sIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAk2Bd,kBAl2BA,SAk2BU,QvB5kIO,QuB4kIjB,C;MAl2BsB,mBAAmB,MAA7B,OrB5sIO,W;K;IqB+sIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAo2Bd,kBAp2BA,SAo2BU,QvB9lIO,QuB8lIjB,C;MAp2BsB,mBAAmB,MAA7B,OrB9sIO,W;K;IqBitIX,4C;MAMI,IAAI,mBAAJ,C;QAAe,OAAO,S;MACD,kBAs2Bd,kBAt2BA,SAs2BU,QvBxmIO,QuBwmIjB,C;MAt2BsB,mBAAmB,MAA7B,OrBxtIO,

iBAAQ,EAAO,OAAP,CAAnB,C;O;KAXX,C;oFAcA,yB;MAAA,gD;MAAA,2C;MAAA,qC;QAWI,OAAO,qBAAoB,OAAR,iBAAQ,EAAO,OAAP,CAApB,C;O;KAXX,C;4FAcA,yB;MAAA,0D;MAAA,uC;MAAA,gD;QAaI,OAAO,mBAAkB,YAAR,iBAAQ,EAAY,SAAZ,EAAuB,OAAvB,CAAlB,C;O;KAbX,C;8FAgBA,yB;MAAA,0D;MAAA,yC;MAAA,gD;QAaI,OAAO,oBAAmB,YAAR,iBAAQ,EAAY,SAAZ,EAAuB,OAAvB,CAAnB,C;O;KAbX,C;8FAgBA,yB;MAAA,0D;MAAA,yC;MAAA,gD;QAaI,OAAO,oBAAmB,YAAR,iBAAQ,EAAY,SAAZ,EAAuB,OAAvB,CAAnB,C;O;KAbX,C;6FAgBA,yB;MAAA,0D;MAAA,2C;MAAA,gD;QAaI,OAAO,qBAAoB,YAAR,iBAAQ,EAAY,SAAZ,EAAuB,OAAvB,CAApB,C;O;KAbX,C;IAgBA,sD;MAAW,yC,yB;QAAA,YAAiB,C;MAAjB,gG,uB;QAAA,YAAiB,C;MAAjB,gG,c;MAAA,gE,c

B,EAAsC,cAAtC,C;MACb,YAAU,SAAV,EAAgB,SAAhB,EAA2B,OAA3B,C;K;IAGJ,+C;MAa4B,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,c;MAC3D,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EAAsC,cAAtC,C;MACb,YAAU,SAAV,EAAgB,SAAhB,EAA2B,OAA3B,C;K;IAGJ,0D;MAaI,kBAAK,SAAL,EAAgB,OAAhB,EAAyB,C;K;IAAo8CZ,0D;MAAI,kBAAK,SAAL,EAAgB,OAAhB,EAAyB,C;MAAj8CQ,WAAR,iBAAQ,EAAR,iCAAQ,EAAR,iCAAQ,EAAR,kCAAQ,C;K;IAAo8CZ,2D;MAAI,kBAAK,SAAL,EAAgB,OAAhB,EAAyB,C;MAAj8CQ,WAAR,iBAAQ,EAAR,iCAAQ,EAAR,iCAAQ,EAAR,kCAAQ,C;K;IAAo8CZ,0

cAAhB,C;UAAgB,yB;UACZ,WA5DoC,SA4DzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7DhB,OA+DO,W;O;KA3EX,C;qGAeA,yB;MAAA,+D;MA+DA,gD;MA/DA,uC;QAYW,kBAAiB,gB;QA8DR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA/DoC,SA4DzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhEhB,OAkEO,W;O;KA9EX,C;qGAeA,yB;MAAA,+D;MAkEA,gD;MAlEA,uC;QAYW,kBAAiB,gB;QAiER,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAlEoC,SAkEzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAnEhB,OAqEO,W;O;KAjFX,C;qGAeA,yB;MAAA,+D;MAqEA,gD;MArEA,uC;QAYW,kBAAiB,gB;QAoER,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WArEoC,SAqEzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtEhB,OAwEO,W;O;KApFX,C;qGAeA,yB;MAAA,+D;MAwEA,gD;MAxEA,uC;QAYW,kBAAiB,gB;QAuER,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAxEoC,SAwEzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAzEhB,OA2EO,W;O;KAvFX,C;qGAeA,yB;MAAA,+D;MA2EA,gD;MA3EA,uC;QAYW,kBAAiB,gB;QA0ER,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA3EoC,SA2EzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5EhB,OA8EO,W;O;KA1FX,C;qGAeA,yB;MAAA,+D;MA8EA,gD;MA9EA,uC;QAYW,kBAAiB,gB;QA6ER,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA9EoC,SA8EzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/EhB,OAiFO,W;O;KA7FX,C;qGAeA,yB;MAAA,+D;MAiFA,gD;MAjFA,uC;QAYW,kBAAiB,gB;QAgFR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAjFoC,SAiFzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlFhB,OAoFO,W;O;KAhGX,C;qGAeA,yB;MAAA,+D;MAoFA,gD;MApFA,uC;QAYW,kBAAiB,gB;QAmFR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WApFoC,SAoFzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QArFhB,OAuFO,W;O;KAnGX,C;qGAeA,yB;MAAA,+D;MAuFA,gD;MAvFA,uC;QAYW,kBAAiB,gB;QAsFR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAvFoC,SAuFzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxFhB,OA0FO,W;O;KAtGX,C;qGAeA,yB;MAAA,+D;MA0FA,gD;MA1FA,uC;QAYW,kBAAiB,gB;QAyFR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA1FoC,SA0FzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA3FhB,OA6FO,W;O;KAzGX,C;qGAeA,yB;MAAA,+D;MA6FA,gD;MA7FA,uC;QAYW,kBAAiB,gB;QA4FR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA7FoC,SA6FzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9FhB,OAgGO,W;O;KA5GX,C;qGAeA,yB;MAAA,+D;MAgGA,gD;MAhGA,uC;QAYW,kBAAiB,gB;QA+FR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAhGoC,SAgGzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAjGhB,OAmGO,W;O;KA/GX,C;qGAeA,yB;MAAA,+D;MAmGA,gD;MAnGA,uC;QAYW,kBAAiB,gB;QAkGR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAnGoC,SAmGzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApGhB,OAsGO,W;O;KAlHX,C;qGAeA,yB;MAAA,+D;MAsGA,gD;MAtGA,uC;QAYW,kBAAiB,gB;QAqGR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAtGoC,SAsGzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAvGhB,OAyGO,W;O;KArHX,C;qGAeA,yB;MAAA,+D;MAyGA,gD;MAzGA,uC;QAYW,kBAAiB,gB;QAwGR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAzGoC,SAyGzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1GhB,OA4GO,W;O;KAxHX,C;qGAeA,yB;MAAA,+D;MA4GA,gD;MA5GA,uC;QAYW,kBAAiB,gB;QA2GR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA5GoC,SA4GzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7GhB,OA+GO,W;O;KA3HX,C;qGAeA,yB;MAAA,+D;MA+GA,gD;MA/GA,uC;QAYW,kBAAiB,gB;QA8GR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA/GoC,SA+GzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhHhB,OAkHO,W;O;KA9HX,C;qGAeA,yB;MAAA,+D;MAkHA,gD;MAlHA,uC;QAYW,kBAAiB,gB;QAiHR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAlHoC,SAkHzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAnHhB,OAqHO,W;O;KAjIX,C;qGAeA,yB;MAAA,+D;MAqHA,gD;MArHA,uC;QAYW,kBAAiB,gB;QAoHR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WArHoC,SAqHzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtHhB,OAwHO,W;O;KApIX,C;qGAeA,yB;MAAA,+D;MAwHA,gD;MAxHA,uC;QAYW,kBAAiB,gB;QAuHR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAxHoC,SAwHzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAzHhB,OA2HO,W;O;KAvIX,C;qGAeA,yB;MAAA,+D;MA2HA,gD;MA3HA,uC;QAYW,kBAAiB,gB;QA0HR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA3HoC,SA2HzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5HhB,OA8HO,W;O;KA1IX,C;qGAeA,yB;MAAA,+D;MA8HA,gD;MA9HA,uC;QAYW,kBAAiB,gB;QA6HR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA9HoC,SA8HzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/HhB,OAiIO,W;O;KA7IX,C;qGAeA,yB;MAAA,+D;MAiIA,gD;MAjIA,uC;QAYW,kBAAiB,gB;QAgIR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAjIoC,SAiIzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlIhB,OAoIO,W;O;KAhJX,C;qGAeA,yB;MAAA,+D;MAoIA,gD;MApIA,uC;QAYW,kBAAiB,gB;QAmIR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WApIoC,SAoIzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QArIhB,OAuIO,W;O;KAnJX,C;qGAeA,yB;MAAA,+D;MAuIA,gD;MAvIA,uC;QAYW,kBAAiB,gB;QAsIR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAvIoC,SAuIzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxIhB,OA0IO,W;O;KAtJX,C;qGAeA,yB;MAAA,+D;MA0IA,gD;MA1IA,uC;QAYW,kBAAiB,gB;QAyIR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA1IoC,SA0IzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA3IhB,OA6IO,W;O;KAzJX,C;qGAeA,yB;MAAA,+D;MA6IA,gD;MA7IA,uC;QAYW,kBAAiB,gB;QA4IR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA7IoC,SA6IzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9IhB,OAgJO,W;O;KA5JX,C;qGAeA,yB;MAAA,+D;MAgJA,gD;MAhJA,uC;QAYW,kBAAiB,gB;QA+IR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAhJoC,SAgJzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAjJhB,OAmJO,W;O;KA/JX,C;qGAeA,yB;MAAA,+D;MAmJA,gD;MAnJA,uC;QAYW,kBAAiB,gB;QAkJR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAnJoC,SAmJzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApJhB,OAsJO,W;O;KAlKX,C;qGAeA,yB;MAAA,+D;MAsJA,gD;MAtJA,uC;QAYW,kBAAiB,gB;QAqJR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAtJoC,SAsJzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAvJhB,OAyJO,W;O;KArKX,C;qGAeA,yB;MAAA,+D;MAyJA,gD;MAzJA,uC;QAYW,kBAAiB,gB;QAwJR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAzJoC,SAyJzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1JhB,OA4JO,W;O;KAxKX,C;qGAeA,yB;MAAA,+D;MA4JA,gD;MA5JA,uC;QAYW,kBAAiB,gB;QA2JR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA5JoC,SA4JzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7JhB,OA+JO,W;O;KA3KX,C;qGAeA,yB;MAAA,+D;MA+JA,gD;MA/JA,uC;QAYW,kBAAiB,gB;QA8JR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA/JoC,SA+JzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhKhB,OAkKO,W;O;KA9KX,C;qGAeA,yB;MAAA,+D;MAkKA,gD;MAlKA,uC;QAYW,kBAAiB,gB;QAiKR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAlKoC,SAkKzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAnKhB,OAqKO,W;O;KAjLX,C;qGAeA,yB;MAAA,+D;MAqKA,gD;MArKA,uC;QAYW,kBAAiB,gB;QAoKR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WArKoC,SAqKzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtKhB,OAwKO,W;O;KApLX,C;qGAeA,yB;MAAA,+D;MAwKA,gD;MAxKA,uC;QAYW,kBAAiB,gB;QAuKR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAxKoC,SAwKzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAzKhB,OA2KO,W;O;KAvLX,C;qGAeA,yB;MAAA,+D;MA2KA,gD;MA3KA,uC;QAYW,kBAAiB,gB;QA0KR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA3KoC,SA2KzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5KhB,OA8KO,W;O;KA1LX,C;qGAeA,yB;MAAA,+D;MA8KA,gD;MA9KA,uC;QAYW,kBAAiB,gB;QA6KR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA9KoC,SA8KzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/KhB,OAiLO,W;O;KA7LX,C;qGAeA,yB;MAAA,+D;MAiLA,gD;MAjLA,uC;QAYW,kBAAiB,gB;QAgLR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAjLoC,SAiLzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlLhB,OAoLO,W;O;KAhMX,C;qGAeA,yB;MAAA,+D;MAoLA,gD;MApLA,uC;QAYW,kBAAiB,gB;QAmLR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WApLoC,SAoLzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QArLhB,OAuLO,W;O;KAnMX,C;qGAeA,yB;MAAA,+D;MAuLA,gD;MAvLA,uC;QAYW,kBAAiB,gB;QAsLR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAvLoC,SAuLzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxLhB,OA0LO,W;O;KAtMX,C;qGAeA,yB;MAAA,+D;MA0LA,gD;MA1LA,uC;QAYW,kBAAiB,gB;QAyLR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA1LoC,SA0LzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA3LhB,OA6LO,W;O;KAzMX,C;qGAeA,yB;MAAA,+D;MA6LA,gD;MA7LA,uC;QAYW,kBAAiB,gB;QA4LR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA7LoC,SA6LzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9LhB,OAgMO,W;O;KA5MX,C;qGAeA,yB;MAAA,+D;MAgMA,gD;MAhMA,uC;QAYW,kBAAiB,gB;QA+LR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAhMoC,SAgMzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAjMhB,OAmMO,W;O;KA/MX,C;qGAeA,yB;MAAA,+D;MAmMA,gD;MAnMA,uC;QAYW,kBAAiB,gB;QAkMR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAnMoC,SAmMzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApMhB,OAsMO,W;O;KAlNX,C;qGAeA,yB;MAAA,+D;MAsMA,gD;MAtMA,uC;QAYW,kBAAiB,gB;QAqMR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAtMoC,SAsMzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAvMhB,OAyMO,W;O;KArNX,C;qGAeA,yB;MAAA,+D;MAyMA,gD;MAzMA,uC;QAYW,kBAAiB,gB;QAwMR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAzMoC,SAyMzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1MhB,OA4MO,W;O;KAxNX,C;qGAeA,yB;MAAA,+D;MA4MA,gD;MA5MA,uC;QAYW,kBAAiB,gB;QA2MR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA5MoC,SA4MzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7MhB,OA+MO,W;O;KA3NX,C;qGAeA,yB;MAAA,+D;MA+MA,gD;MA/MA,uC;QAYW,kBAAiB,gB;QA8MR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA/MoC,SA+MzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhNhB,OAkNO,W;O;KA9NX,C;qGAeA,yB;MAAA,+D;MAkNA,gD;MAlNA,uC;QAYW,kBAAiB,gB;QAiNR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAlNoC,SAkNzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAnNhB,OAqNO,W;O;KAjOX,C;qGAeA,yB;MAAA,+D;MAqNA,gD;MArNA,uC;QAYW,kBAAiB,gB;QAoNR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WArNoC,SAqNzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtNhB,OAwNO,W;O;KApOX,C;qGAeA,yB;MAAA,+D;MAwNA,gD;MAxNA,uC;QAYW,kBAAiB,gB;QAuNR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAxNoC,SAwNzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAzNhB,OA2NO,W;O;KAvOX,C;qGAeA,yB;MAAA,+D;MA2NA,gD;MA3NA,uC;QAYW,kBAAiB,gB;QA0NR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA3NoC,SA2NzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5NhB,OA8NO,W;O;KA1OX,C;qGAeA,yB;MAAA,+D;MA8NA,gD;MA9NA,uC;QAYW,kBAAiB,gB;QA6NR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA9NoC,SA8NzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/NhB,OAiOO,W;O;KA7OX,C;qGAeA,yB;MAAA,+D;MAiOA,gD;MAjOA,uC;QAYW,kBAAiB,gB;QAgOR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAjOoC,SAiOzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlOhB,OAoOO,W;O;KAhPX,C;qGAeA,yB;MAAA,+D;MAoOA,gD;MApOA,uC;QAYW,kBAAiB,gB;QAmOR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WApOoC,SAoOzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QArOhB,OAuOO,W;O;KAnPX,C;qGAeA,yB;MAAA,+D;MAuOA,gD;MAvOA,uC;QAYW,kBAAiB,gB;QAsOR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAvOoC,SAuOzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxOhB,OA0OO,W;O;KAtPX,C;qGAeA,yB;MAAA,+D;MA0OA,gD;MA1OA,uC;QAYW,kBAAiB,gB;QAyOR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA1OoC,SA0OzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA3OhB,OA6OO,W;O;KAzPX,C;qGAeA,yB;MAAA,+D;MA6OA,gD;MA7OA,uC;QAYW,kBAAiB,gB;QA4OR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA7OoC,SA6OzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA9OhB,OAgPO,W;O;KA5PX,C;qGAeA,yB;MAAA,+D;MAgPA,gD;MAhPA,uC;QAYW,kBAAiB,gB;QA+OR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAhPoC,SAgPzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAjPhB,OAmPO,W;O;KA/PX,C;qGAeA,yB;MAAA,+D;MAmPA,gD;MAnPA,uC;QAYW,kBAAiB,gB;QAkPR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAnPoC,SAmPzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QApPhB,OAsPO,W;O;KAlQX,C;qGAeA,yB;MAAA,+D;MAsPA,gD;MAtPA,uC;QAYW,kBAAiB,gB;QAqPR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAtPoC,SAsPzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAvPhB,OAyPO,W;O;KArQX,C;qGAeA,yB;MAAA,+D;MAyPA,gD;MAzPA,uC;QAYW,kBAAiB,gB;QAwPR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAzPoC,SAyPzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA1PhB,OA4PO,W;O;KAxQX,C;qGAeA,yB;MAAA,+D;MA4PA,gD;MA5PA,uC;QAYW,kBAAiB,gB;QA2PR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA5PoC,SA4PzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7PhB,OA+PO,W;O;KA3QX,C;qGAeA,yB;MAAA,+D;MA+PA,gD;MA/PA,uC;QAYW,kBAAiB,gB;QA8PR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA/PoC,SA+PzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAhQhB,OAkQO,W;O;KA9QX,C;qGAeA,yB;MAAA,+D;MAkQA,gD;MAlQA,uC;QAYW,kBAAiB,gB;QAiQR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAlQoC,SAkQzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAnQhB,OAqQO,W;O;KAjRX,C;qGAeA,yB;MAAA,+D;MAqQA,gD;MArQA,uC;QAYW,kBAAiB,gB;QAoQR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WArQoC,SAqQzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAtQhB,OAwQO,W;O;KApRX,C;qGAeA,yB;MAAA,+D;MAwQA,gD;MAxQA,uC;QAYW,kBAAiB,gB;QAuQR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAxQoC,SAwQzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAzQhB,OA2QO,W;O;KAvRX,C;qGAeA,yB;MAAA,+D;MA2QA,gD;MA3QA,uC;QAYW,kBAAiB,gB;QA0QR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA3QoC,SA2QzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA5QhB,OA8QO,W;O;KA1RX,C;qGAeA,yB;MAAA,+D;MA8QA,gD;MA9QA,uC;QAYW,kBAAiB,gB;QA6QR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA9QoC,SA8QzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA/QhB,OAiRO,W;O;KA7RX,C;qGAeA,yB;MAAA,+D;MAiRA,gD;MAjRA,uC;QAYW,kBAAiB,gB;QAgRR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAjRoC,SAiRzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAlRhB,OAoRO,W;O;KAhSX,C;qGAeA,yB;MAAA,+D;MAoRA,gD;MApRA,uC;QAYW,kBAAiB,gB;QAmRR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WApRoC,SAoRzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QArRhB,OAuRO,W;O;KAnSX,C;qGAeA,yB;MAAA,+D;MAuRA,gD;MAvRA,uC;QAYW,kBAAiB,gB;QAsRR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAvRoC,SAuRzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAxRhB,OA0RO,W;O;KAtSX,C;qGAeA,yB;MAAA,+D;MA0RA,gD;MA1RA,uC;QAYW,kBAAiB,gB;QAyRR,gB;QADhB,YAAY,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WA1RoC,SA0RzB,EAAU,cAAV,EAAU,sBAAV,WAAmB,OAAnB,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA3RhB,OA6RO,W;O;KAzSX,C;qGAeA,yB;MAAA,+D;MA6RA,gD;MA7RA,uC;QAYW,kBAAiB,gB;QA4RR,gB;QADhB,YAAY,

BjkJhC,EAAS,MAAT,C;YACA,e;;YAEA,c;;U+B8jJA,iB;UACA,IAAK,WAxJyD,cAwJrD,CAAe,OAAf,CAAJ,C;;QAxJT,OA0JO,W;O;KAvKX,C;uFAgBA,yB;MAAA,wE;MA0JA,+D;MA1JA,yD;QAaW,kBAAU,oB;QA0JD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA3JiD,WA2JvC,CAAY,OAAZ,C;U/BvlJP,U;UADP,Y+BylJe,W/BzlJH,W+BylJwB,G/BzlJxB,C;UACL,IAAI,aAAJ,C;YACH,a+BulJuC,gB;YAA5B,W/BtlJX,a+BslJgC,G/BtlJhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BmlJA,iB;UACA,IAAK,WA7JyD,cA6JrD,CAAe,OAAf,CAAJ,C;;QA7JT,OA+JO,W;O;KA5KX,C;uFAgBA,yB;MAAA,wE;MA+JA,+D;MA/JA,yD;QAaW,kBAAU,oB;QA+JD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAhKiD,WAgKvC,CAAY,OAAZ,C;U/B5kJP,U;UADP,Y+B8kJe,W/B9kJH,W+B8kJwB,G/B9kJxB,C;UACL,IAAI,aAAJ,C;YACH,a+B4kJuC,gB;YAA5B,W/B3kJX,a+B2kJgC,G/B3kJhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BwkJA,iB;UACA,IAAK,WAlKyD,cAkKrD,CAAe,OAAf,CAAJ,C;;QAlKT,OAoKO,W;O;KAjLX,C;uFAgBA,yB;MAAA,wE;MAoKA,+D;MApKA,yD;QAaW,kBAAU,oB;QAoKD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UArKiD,WAqKvC,CAAY,OAAZ,C;U/BjkJP,U;UADP,Y+BmkJe,W/BnkJH,W+BmkJwB,G/BnkJxB,C;UACL,IAAI,aAAJ,C;YACH,a+BikJuC,gB;YAA5B,W/BhkJX,a+BgkJgC,G/BhkJhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B6jJA,iB;UACA,IAAK,WAvKyD,cAuKrD,CAAe,OAAf,CAAJ,C;;QAvKT,OAyKO,W;O;KAtLX,C;uFAgBA,yB;MAAA,wE;MAyKA,+D;MAzKA,yD;QAaW,kBAAU,oB;QAyKD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA1KiD,WA0KvC,CAAY,OAAZ,C;U/BtjJP,U;UADP,Y+BwjJe,W/BxjJH,W+BwjJwB,G/BxjJxB,C;UACL,IAAI,aAAJ,C;YACH,a+BsjJuC,gB;YAA5B,W/BrjJX,a+BqjJgC,G/BrjJhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BkjJA,iB;UACA,IAAK,WA5KyD,cA4KrD,CAAe,OAAf,CAAJ,C;;QA5KT,OA8KO,W;O;KA3LX,C;uFAgBA,yB;MAAA,wE;MA8KA,+D;MA9KA,yD;QAaW,kBAAU,oB;QA8KD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA/KiD,WA+KvC,CAAY,OAAZ,C;U/B3iJP,U;UADP,Y+B6iJe,W/B7iJH,W+B6iJwB,G/B7iJxB,C;UACL,IAAI,aAAJ,C;YACH,a+B2iJuC,gB;YAA5B,W/B1iJX,a+B0iJgC,G/B1iJhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BuiJA,iB;UACA,IAAK,WAjLyD,cAiLrD,CAAe,OAAf,CAAJ,C;;QAjLT,OAmLO,W;O;KAhMX,C;uFAgBA,yB;MAAA,wE;MAmLA,+D;MAnLA,yD;QAaW,kBAAU,oB;QAmLD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UApLiD,WAoLvC,CAAY,OAAZ,C;U/BhiJP,U;UADP,Y+BkiJe,W/BliJH,W+BkiJwB,G/BliJxB,C;UACL,IAAI,aAAJ,C;YACH,a+BgiJuC,gB;YAA5B,W/B/hJX,a+B+hJgC,G/B/hJhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B4hJA,iB;UACA,IAAK,WAtLyD,cAsLrD,CAAe,OAAf,CAAJ,C;;QAtLT,OAwLO,W;O;KArMX,C;uFAgBA,yB;MAAA,wE;MAwLA,+D;MAxLA,yD;QAaW,kBAAU,oB;QAwLD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAzLiD,WAyLvC,CAAY,OAAZ,C;U/B7hJP,U;UADP,Y+B+hJe,W/B/hJH,W+B+hJwB,G/B/hJxB,C;UACL,IAAI,aAAJ,C;YACH,a+B6hJuC,gB;YAA5B,W/B5hJX,a+B4hJgC,G/B5hJhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+ByhJA,iB;UACA,IAAK,WA3LyD,cA2LrD,CAAe,OAAf,CAAJ,C;;QA3LT,OA6LO,W;O;KA1MX,C;uFAgBA,yB;MAAA,wE;MA6LA,+D;MA7LA,yD;QAaW,kBAAU,oB;QA6LD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA9LiD,WA8LvC,CAAY,OAAZ,C;U/B1gJP,U;UADP,Y+B4gJe,W/B5gJH,W+B4gJwB,G/B5gJxB,C;UACL,IAAI,aAAJ,C;YACH,a+B0gJuC,gB;YAA5B,W/BzgJX,a+BygJgC,G/BzgJhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BsgJA,iB;UACA,IAAK,WAhMyD,cAgMrD,CAAe,OAAf,CAAJ,C;;QAhMT,OAkMO,W;O;KA/MX,C;uFAgBA,yB;MAAA,wE;MAkMA,+D;MAlMA,yD;QAaW,kBAAU,oB;QAkMD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAnMiD,WAmMvC,CAAY,OAAZ,C;U/BvfP,U;UADP,Y+ByfO,W/BzfK,W+ByfY,G/BzfZ,C;UACL,IAAI,aAAJ,C;YACH,a+BufuC,gB;YAA5B,W/BtfX,a+BsfgC,G/BtfhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BmfA,iB;UACA,IAAK,WArMyD,cAqMrD,CAAe,OAAf,CAAJ,C;;QArMT,OAuMO,W;O;KApNX,C;uFAgBA,yB;MAAA,wE;MAuMA,+D;MAvMA,yD;QAaW,kBAAU,oB;QAuMD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAxMiD,WAwMvC,CAAY,OAAZ,C;U/BpeP,U;UADP,Y+BseO,W/BteK,W+BseY,G/BteZ,C;UACL,IAAI,aAAJ,C;YACH,a+BoeuC,gB;YAA5B,W/BneX,a+BmegC,G/BnehC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BgeA,iB;UACA,IAAK,WA1MyD,cA0MrD,CAAe,OAAf,CAAJ,C;;QA1MT,OA4MO,W;O;KAzNX,C;uFAgBA,yB;MAAA,wE;MA4MA,+D;MA5MA,yD;QAaW,kBAAU,oB;QA4MD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA7MiD,WA6MvC,CAAY,OAAZ,C;U/BjdP,U;UADP,Y+BmdO,W/BndK,W+BmdY,G/BndZ,C;UACL,IAAI,aAAJ,C;YACH,a+BidiC,gB;YAA5B,W/BhdX,a+BgdgC,G/BhdhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B6cA,iB;UACA,IAAK,WA/MyD,cA+MrD,CAAe,OAAf,CAAJ,C;;QA/MT,OAiNO,W;O;KA9NX,C;uFAgBA,yB;MAAA,wE;MAiNA,+D;MAjNA,yD;QAaW,kBAAU,oB;QAiND,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAlNiD,WAkNvC,CAAY,OAAZ,C;U/B3cP,U;UADP,Y+B6cO,W/B7cK,W+B6cY,G/B7cZ,C;UACL,IAAI,aAAJ,C;YACH,a+B2cuC,gB;YAA5B,W/B1cX,a+B0cgC,G/B1chC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BucA,iB;UACA,IAAK,WAzNyD,cAyNrD,CAAe,OAAf,CAAJ,C;;QAzNT,OA2NO,W;O;KAxOX,C;uFAgBA,yB;MAAA,wE;MA2NA,+D;MA3NA,yD;QAaW,kBAAU,oB;QA2ND,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA5NiD,WA4NvC,CAAY,OAAZ,C;U/BrbP,U;UADP,Y+BubO,W/BvbK,W+BubY,G/BvbZ,C;UACL,IAAI,aAAJ,C;YACH,a+BqbuC,gB;YAA5B,W/BpbX,a+BobgC,G/BpbhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BibA,iB;UACA,IAAK,WAnOyD,cAmOrD,CAAe,OAAf,CAAJ,C;;QAnOT,OAqOO,W;O;KAlPX,C;uFAgBA,yB;MAAA,wE;MAqOA,+D;MArOA,yD;QAaW,kBAAU,oB;QAqOD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAtOiD,WAsOvC,CAAY,OAAZ,C;U/B/aP,U;UADP,Y+BibO,W/BjbK,W+BibY,G/BjbZ,C;UACL,IAAI,aAAJ,C;YACH,a+B+auC,gB;YAA5B,W/B9aX,a+B8agC,G/B9ahC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B2aA,iB;UACA,IAAK,WAvOyD,cAuOrD,CAAe,OAAf,CAAJ,C;;QAvOT,OAyOO,W;O;KAtPX,C;uFAgBA,yB;MAAA,wE;MAyOA,+D;MAzOA,yD;QAaW,kBAAU,oB;QAyOD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA1OiD,WA0OvC,CAAY,OAAZ,C;U/BzZP,U;UADP,Y+B2ZO,W/B3ZK,W+B2ZY,G/B3ZZ,C;UACL,IAAI,aAAJ,C;YACH,a+ByZuC,gB;YAA5B,W/BxZX,a+BwZgC,G/BxZhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BqZA,iB;UACA,IAAK,WA3OyD,cA2OrD,CAAe,OAAf,CAAJ,C;;QA3OT,OA6OO,W;O;KA1PX,C;uFAgBA,yB;MAAA,wE;MA6OA,+D;MA7OA,yD;QAaW,kBAAU,oB;QA6OD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA9OiD,WA8OvC,CAAY,OAAZ,C;U/BnYP,U;UADP,Y+BqYO,W/BrYK,W+BqYY,G/BrYZ,C;UACL,IAAI,aAAJ,C;YACH,a+BmYuC,gB;YAA5B,W/BlYX,a+BkYgC,G/BlYhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B+XA,iB;UACA,IAAK,WA/OyD,cA+OrD,CAAe,OAAf,CAAJ,C;;QA/OT,OAiPO,W;O;KA9PX,C;uFAgBA,yB;MAAA,wE;MAiPA,+D;MAjPA,yD;QAaW,kBAAU,oB;QAiPD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAlPiD,WAkPvC,CAAY,OAAZ,C;U/B7WP,U;UADP,Y+B+WO,W/B/WK,W+B+WY,G/B/WZ,C;UACL,IAAI,aAAJ,C;YACH,a+B6WuC,gB;YAA5B,W/B5WX,a+B4WgC,G/B5WhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+ByWA,iB;UACA,IAAK,WAnPyD,cAmPrD,CAAe,OAAf,CAAJ,C;;QAnPT,OAqPO,W;O;KAlQX,C;uFAgBA,yB;MAAA,wE;MAqPA,+D;MArPA,yD;QAaW,kBAAU,oB;QAqPD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAtPiD,WAsPvC,CAAY,OAAZ,C;U/BjVP,U;UADP,Y+BmVO,W/BnVK,W+BmVY,G/BnVZ,C;UACL,IAAI,aAAJ,C;YACH,a+BiVuC,gB;YAA5B,W/BhVX,a+BgVgC,G/BhVhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B6UA,iB;UACA,IAAK,WAvPyD,cAuPrD,CAAe,OAAf,CAAJ,C;;QAvPT,OAyPO,W;O;KAtQX,C;uFAgBA,yB;MAAA,wE;MAyPA,+D;MAzPA,yD;QAaW,kBAAU,oB;QAyPD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA1PiD,WA0PvC,CAAY,OAAZ,C;U/BzTP,U;UADP,Y+B2TO,W/B3TK,W+B2TY,G/B3TZ,C;UACL,IAAI,aAAJ,C;YACH,a+ByTuC,gB;YAA5B,W/BxTX,a+BwTgC,G/BxThC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BqTA,iB;UACA,IAAK,WA3PyD,cA2PrD,CAAe,OAAf,CAAJ,C;;QA3PT,OA6PO,W;O;KA1QX,C;uFAgBA,yB;MAAA,wE;MA6PA,+D;MA7PA,yD;QAaW,kBAAU,oB;QA6PD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA9PiD,WA8PvC,CAAY,OAAZ,C;U/BnSP,U;UADP,Y+BqSO,W/BrSK,W+BqSY,G/BrSZ,C;UACL,IAAI,aAAJ,C;YACH,a+BmSuC,gB;YAA5B,W/BlSX,a+BkSgC,G/BlShC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B+RA,iB;UACA,IAAK,WA/PyD,cA+PrD,CAAe,OAAf,CAAJ,C;;QA/PT,OAiQO,W;O;KA9QX,C;uFAgBA,yB;MAAA,wE;MAiQA,+D;MAjQA,yD;QAaW,kBAAU,oB;QAiQD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAlQiD,WAkQvC,CAAY,OAAZ,C;U/B7QP,U;UADP,Y+B+QO,W/B/QK,W+B+QY,G/B/QZ,C;UACL,IAAI,aAAJ,C;YACH,a+B6QuC,gB;YAA5B,W/B5QX,a+B4QgC,G/B5QhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+ByQA,iB;UACA,IAAK,WAnQyD,cAmQrD,CAAe,OAAf,CAAJ,C;;QAnQT,OAqQO,W;O;KAlRX,C;uFAgBA,yB;MAAA,wE;MAqQA,+D;MArQA,yD;QAaW,kBAAU,oB;QAqQD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAtQiD,WAsQvC,CAAY,OAAZ,C;U/BjPP,U;UADP,Y+BmPO,W/BnPK,W+BmPY,G/BnPZ,C;UACL,IAAI,aAAJ,C;YACH,a+BiPuC,gB;YAA5B,W/BhPX,a+BgPgC,G/BhPhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B6OA,iB;UACA,IAAK,WAvQyD,cAuQrD,CAAe,OAAf,CAAJ,C;;QAvQT,OAyQO,W;O;KAtRX,C;uFAgBA,yB;MAAA,wE;MAyQA,+D;MAzQA,yD;QAaW,kBAAU,oB;QAyQD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA1QiD,WA0QvC,CAAY,OAAZ,C;U/BzOP,U;UADP,Y+B2OO,W/B3OK,W+B2OY,G/B3OZ,C;UACL,IAAI,aAAJ,C;YACH,a+ByOuC,gB;YAA5B,W/BxOX,a+BwOgC,G/BxOhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BqOA,iB;UACA,IAAK,WA3QyD,cA2QrD,CAAe,OAAf,CAAJ,C;;QA3QT,OA6QO,W;O;KA1RX,C;uFAgBA,yB;MAAA,wE;MA6QA,+D;MA7QA,yD;QAaW,kBAAU,oB;QA6QD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA9QiD,WA8QvC,CAAY,OAAZ,C;U/BnPP,U;UADP,Y+BqPO,W/BrPK,W+BqPY,G/BrPZ,C;UACL,IAAI,aAAJ,C;YACH,a+BmPuC,gB;YAA5B,W/BlPX,a+BkPgC,G/BlPhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B+OA,iB;UACA,IAAK,WA/QyD,cA+QrD,CAAe,OAAf,CAAJ,C;;QA/QT,OAiRO,W;O;KA9RX,C;uFAgBA,yB;MAAA,wE;MAiRA,+D;MAjRA,yD;QAaW,kBAAU,oB;QAiRD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAlRiD,WAkRvC,CAAY,OAAZ,C;U/B7PP,U;UADP,Y+B+PO,W/B/PK,W+B+PY,G/B/PZ,C;UACL,IAAI,aAAJ,C;YACH,a+B6PuC,gB;YAA5B,W/B5PX,a+B4PgC,G/B5PhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+ByPA,iB;UACA,IAAK,WAnRyD,cAmRrD,CAAe,OAAf,CAAJ,C;;QAnRT,OAqRO,W;O;KAlSX,C;uFAgBA,yB;MAAA,wE;MAqRA,+D;MArRA,yD;QAaW,kBAAU,oB;QAqRD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAtRiD,WAsRvC,CAAY,OAAZ,C;U/BjQP,U;UADP,Y+BmQO,W/BnQK,W+BmQY,G/BnQZ,C;UACL,IAAI,aAAJ,C;YACH,a+BiQuC,gB;YAA5B,W/BhQX,a+BgQgC,G/BhQhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B6PA,iB;UACA,IAAK,WAvRyD,cAuRrD,CAAe,OAAf,CAAJ,C;;QAvRT,OAyRO,W;O;KAtSX,C;uFAgBA,yB;MAAA,wE;MAyRA,+D;MAzRA,yD;QAaW,kBAAU,oB;QAyRD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA1RiD,WA0RvC,CAAY,OAAZ,C;U/BzQP,U;UADP,Y+B2QO,W/B3QK,W+B2QY,G/B3QZ,C;UACL,IAAI,aAAJ,C;YACH,a+ByQuC,gB;YAA5B,W/BxQX,a+BwQgC,G/BxQhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BqQA,iB;UACA,IAAK,WA3RyD,cA2RrD,CAAe,OAAf,CAAJ,C;;QA3RT,OA6RO,W;O;KA1SX,C;uFAgBA,yB;MAAA,wE;MA6RA,+D;MA7RA,yD;QAaW,kBAAU,oB;QA6RD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA9RiD,WA8RvC,CAAY,OAAZ,C;U/BnRP,U;UADP,Y+BqRO,W/BrRK,W+BqRY,G/BrRZ,C;UACL,IAAI,aAAJ,C;YACH,a+BmRuC,gB;YAA5B,W/BlRX,a+BkRgC,G/BlRhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B+QA,iB;UACA,IAAK,WA/RyD,cA+RrD,CAAe,OAAf,CAAJ,C;;QA/RT,OAiSO,W;O;KA9SX,C;uFAgBA,yB;MAAA,wE;MAiSA,+D;MAjSA,yD;QAaW,kBAAU,oB;QAiSD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAlSiD,WAkSvC,CAAY,OAAZ,C;U/B7RP,U;UADP,Y+B+RO,W/B/RK,W+B+RY,G/B/RZ,C;UACL,IAAI,aAAJ,C;YACH,a+B6RuC,gB;YAA5B,W/B5RX,a+B4RgC,G/B5RhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+ByRA,iB;UACA,IAAK,WAnSyD,cAmSrD,CAAe,OAAf,CAAJ,C;;QAnST,OAqSO,W;O;KAlTX,C;uFAgBA,yB;MAAA,wE;MAqSA,+D;MArSA,yD;QAaW,kBAAU,oB;QAqSD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAtSiD,WAsSvC,CAAY,OAAZ,C;U/BjSP,U;UADP,Y+BmSO,W/BnSK,W+BmSY,G/BnSZ,C;UACL,IAAI,aAAJ,C;YACH,a+BiSuC,gB;YAA5B,W/BhSX,a+BgSgC,G/BhShC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B6RA,iB;UACA,IAAK,WAvSyD,cAuSrD,CAAe,OAAf,CAAJ,C;;QAvST,OAySO,W;O;KAtTX,C;uFAgBA,yB;MAAA,wE;MAySA,+D;MAzSA,yD;QAaW,kBAAU,oB;QAySD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA1SiD,WA0SvC,CAAY,OAAZ,C;U/BzSP,U;UADP,Y+B2SO,W/B3SK,W+B2SY,G/B3SZ,C;UACL,IAAI,aAAJ,C;YACH,a+BySuC,gB;YAA5B,W/BxSX,a+BwSgC,G/BxShC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BqSA,iB;UACA,IAAK,WA3SyD,cA2SrD,CAAe,OAAf,CAAJ,C;;QA3ST,OA6SO,W;O;KA1TX,C;uFAgBA,yB;MAAA,wE;MA6SA,+D;MA7SA,yD;QAaW,kBAAU,oB;QA6SD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA9SiD,WA8SvC,CAAY,OAAZ,C;U/BnSP,U;UADP,Y+BqSO,W/BrSK,W+BqSY,G/BrSZ,C;UACL,IAAI,aAAJ,C;YACH,a+BmSuC,gB;YAA5B,W/BlSX,a+BkSgC,G/BlShC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B+RA,iB;UACA,IAAK,WA/SyD,cA+SrD,CAAe,OAAf,CAAJ,C;;QA/ST,OAiTO,W;O;KA9TX,C;uFAgBA,yB;MAAA,wE;MAiTA,+D;MAjTA,yD;QAaW,kBAAU,oB;QAiTD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAlTiD,WAkTvC,CAAY,OAAZ,C;U/B7SP,U;UADP,Y+B+SO,W/B/SK,W+B+SY,G/B/SZ,C;UACL,IAAI,aAAJ,C;YACH,a+B6SuC,gB;YAA5B,W/B5SX,a+B4SgC,G/B5ShC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BySA,iB;UACA,IAAK,WAnTyD,cAmTrD,CAAe,OAAf,CAAJ,C;;QAnTT,OAqTO,W;O;KAlUX,C;uFAgBA,yB;MAAA,wE;MAqTA,+D;MArTA,yD;QAaW,kBAAU,oB;QAqTD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAtTiD,WAsTvC,CAAY,OAAZ,C;U/BjTP,U;UADP,Y+BmTO,W/BnTK,W+BmTY,G/BnTZ,C;UACL,IAAI,aAAJ,C;YACH,a+BiTuC,gB;YAA5B,W/BhTX,a+BgTgC,G/BhThC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B6SA,iB;UACA,IAAK,WAvTyD,cAuTrD,CAAe,OAAf,CAAJ,C;;QAvTT,OAyTO,W;O;KAtUX,C;uFAgBA,yB;MAAA,wE;MAyTA,+D;MAzTA,yD;QAaW,kBAAU,oB;QAyTD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA1TiD,WA0TvC,CAAY,OAAZ,C;U/BzTP,U;UADP,Y+B2TO,W/B3TK,W+B2TY,G/B3TZ,C;UACL,IAAI,aAAJ,C;YACH,a+ByTuC,gB;YAA5B,W/BxTX,a+BwTgC,G/BxThC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BqTA,iB;UACA,IAAK,WA3TyD,cA2TrD,CAAe,OAAf,CAAJ,C;;QA3TT,OA6TO,W;O;KA1UX,C;uFAgBA,yB;MAAA,wE;MA6TA,+D;MA7TA,yD;QAaW,kBAAU,oB;QA6TD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA9TiD,WA8TvC,CAAY,OAAZ,C;U/BnTP,U;UADP,Y+BqTO,W/BrTK,W+BqTY,G/BrTZ,C;UACL,IAAI,aAAJ,C;YACH,a+BmTuC,gB;YAA5B,W/BlTX,a+BkTgC,G/BlThC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B+SA,iB;UACA,IAAK,WA/TyD,cA+TrD,CAAe,OAAf,CAAJ,C;;QA/TT,OAiUO,W;O;KA9UX,C;uFAgBA,yB;MAAA,wE;MAiUA,+D;MAjUA,yD;QAaW,kBAAU,oB;QAiUD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAlUiD,WAkUvC,CAAY,OAAZ,C;U/B7TP,U;UADP,Y+B+TO,W/B/TK,W+B+TY,G/B/TZ,C;UACL,IAAI,aAAJ,C;YACH,a+B6TuC,gB;YAA5B,W/B5TX,a+B4TgC,G/B5ThC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+ByTA,iB;UACA,IAAK,WAnUyD,cAmUrD,CAAe,OAAf,CAAJ,C;;QAnUT,OAqUO,W;O;KAlVX,C;uFAgBA,yB;MAAA,wE;MAqUA,+D;MArUA,yD;QAaW,kBAAU,oB;QAqUD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAtUiD,WAsUvC,CAAY,OAAZ,C;U/BjUP,U;UADP,Y+BmUO,W/BnUK,W+BmUY,G/BnUZ,C;UACL,IAAI,aAAJ,C;YACH,a+BiUuC,gB;YAA5B,W/BhUX,a+BgUgC,G/BhUhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B6TA,iB;UACA,IAAK,WAvUyD,cAuUrD,CAAe,OAAf,CAAJ,C;;QAvUT,OAyUO,W;O;KAtVX,C;uFAgBA,yB;MAAA,wE;MAyUA,+D;MAzUA,yD;QAaW,kBAAU,oB;QAyUD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA1UiD,WA0UvC,CAAY,OAAZ,C;U/BzUP,U;UADP,Y+B2UO,W/B3UK,W+B2UY,G/B3UZ,C;UACL,IAAI,aAAJ,C;YACH,a+ByUuC,gB;YAA5B,W/BxUX,a+BwUgC,G/BxUhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BqUA,iB;UACA,IAAK,WA3UyD,cA2UrD,CAAe,OAAf,CAAJ,C;;QA3UT,OA6UO,W;O;KA1VX,C;uFAgBA,yB;MAAA,wE;MA6UA,+D;MA7UA,yD;QAaW,kBAAU,oB;QA6UD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA9UiD,WA8UvC,CAAY,OAAZ,C;U/BnUP,U;UADP,Y+BqUO,W/BrUK,W+BqUY,G/BrUZ,C;UACL,IAAI,aAAJ,C;YACH,a+BmUuC,gB;YAA5B,W/BlUX,a+BkUgC,G/BlUhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B+TA,iB;UACA,IAAK,WA/UyD,cA+UrD,CAAe,OAAf,CAAJ,C;;QA/UT,OAiVO,W;O;KA9VX,C;uFAgBA,yB;MAAA,wE;MAiVA,+D;MAjVA,yD;QAaW,kBAAU,oB;QAiVD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAlViD,WAkVvC,CAAY,OAAZ,C;U/B7UP,U;UADP,Y+B+UO,W/B/UK,W+B+UY,G/B/UZ,C;UACL,IAAI,aAAJ,C;YACH,a+B6UuC,gB;YAA5B,W/B5UX,a+B4UgC,G/B5UhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+ByUA,iB;UACA,IAAK,WAnVyD,cAmVrD,CAAe,OAAf,CAAJ,C;;QAnVT,OAqVO,W;O;KAlWX,C;uFAgBA,yB;MAAA,wE;MAqVA,+D;MArVA,yD;QAaW,kBAAU,oB;QAqVD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAtViD,WAsVvC,CAAY,OAAZ,C;U/BjVP,U;UADP,Y+BmVO,W/BnVK,W+BmVY,G/BnVZ,C;UACL,IAAI,aAAJ,C;YACH,a+BiVuC,gB;YAA5B,W/BhVX,a+BgVgC,G/BhVhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B6UA,iB;UACA,IAAK,WAvVyD,cAuVrD,CAAe,OAAf,CAAJ,C;;QAvVT,OAyVO,W;O;KAtWX,C;uFAgBA,yB;MAAA,wE;MAyVA,+D;MAzVA,yD;QAaW,kBAAU,oB;QAyVD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA1ViD,WA0VvC,CAAY,OAAZ,C;U/BzVP,U;UADP,Y+B2VO,W/B3VK,W+B2VY,G/B3VZ,C;UACL,IAAI,aAAJ,C;YACH,a+ByVuC,gB;YAA5B,W/BxVX,a+BwVgC,G/BxVhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+BqVA,iB;UACA,IAAK,WA3VyD,cA2VrD,CAAe,OAAf,CAAJ,C;;QA3VT,OA6VO,W;O;KA1WX,C;uFAgBA,yB;MAAA,wE;MA6VA,+D;MA7VA,yD;QAaW,kBAAU,oB;QA6VD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA9ViD,WA8VvC,CAAY,OAAZ,C;U/BnVP,U;UADP,Y+BqVO,W/BrVK,W+BqVY,G/BrVZ,C;UACL,IAAI,aAAJ,C;YACH,a+BmVuC,gB;YAA5B,W/BlVX,a+BkVgC,G/BlVhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B+UA,iB;UACA,IAAK,WA/VyD,cA+VrD,CAAe,OAAf,CAAJ,C;;QA/VT,OAiWO,W;O;KA9WX,C;uFAgBA,yB;MAAA,wE;MAiWA,+D;MAjWA,yD;QAaW,kBAAU,oB;QAiWD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAlWiD,WAkWvC,CAAY,OAAZ,C;U/B7VP,U;UADP,Y+B+VO,W/B/VK,W+B+VY,G/B/VZ,C;UACL,IAAI,aAAJ,C;YACH,a+B6VuC,gB;YAA5B,W/B5VX,a+B4VgC,G/B5VhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+ByVA,iB;UACA,IAAK,WAnWyD,cAmWrD,CAAe,OAAf,CAAJ,C;;QAnWT,OAqWO,W;O;KAlXX,C;uFAgBA,yB;MAAA,wE;MAqWA,+D;MArWA,yD;QAaW,kBAAU,oB;QAqWD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAtWiD,WAsWvC,CAAY,OAAZ,C;U/BjWP,U;UADP,Y+BmWO,W/BnWK,W+BmWY,G/BnWZ,C;UACL,IAAI,aAAJ,C;YACH,a+BiWuC,gB;YAA5B,W/BhWX,a+BgWgC,G/BhWhC,C;UACL,IAAI,aAAJ,C;YACH,e;;YAEA,c;;U+B6VA,iB;UACA,IAAK,WAvWyD,cAuWrD,CAAe,OAAf,CAAJ,C;;QAvWT,OAyWO,W;O;KAtXX,C;uFAgBA,yB;MAAA,wE;MAyWA,

wDP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAzDwB,SAyDpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAzDhB,OA0DO,W;O;KApEX,C;6FAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA2DP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA5DwB,SA4DpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA5DhB,OA6DO,W;O;KAvEX,C;4FAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA8DP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA/DwB,SA+DpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA/DhB,OAgEO,W;O;KA1EX,C;iGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAiEP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAlEwB,SAkEpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAlEhB,OAmEO,W;O;KA7EX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAoEP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WArEwB,SAqEpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QArEhB,OAsEO,W;O;KAhFX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAuEP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAxEwB,SAwEpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAxEhB,OAyEO,W;O;KAnFX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA0EP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA3EwB,SA2EpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA3EhB,OA4EO,W;O;KAtFX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA6EP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA9EwB,SA8EpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA9EhB,OA+EO,W;O;KAzFX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAgFP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAjFwB,SAiFpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAjFhB,OAkFO,W;O;KA5FX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAmFP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WApFwB,SAoFpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QApFhB,OAqFO,W;O;KA/FX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAsFP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAvFwB,SAuFpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAvFhB,OAwFO,W;O;KAlGX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAyFP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA1FwB,SA0FpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA1FhB,OA2FO,W;O;KArGX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA4FP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA7FwB,SA6FpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA7FhB,OA8FO,W;O;KAxGX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA+FP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAhGwB,SAgGpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAhGhB,OAiGO,W;O;KA3GX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAkGP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAnGwB,SAmGpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAnGhB,OAoGO,W;O;KA9GX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAqGP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAtGwB,SAsGpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAtGhB,OAuGO,W;O;KAjHX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAwGP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAzGwB,SAyGpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAzGhB,OA0GO,W;O;KApHX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA2GP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA5GwB,SA4GpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA5GhB,OA6GO,W;O;KAvHX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA8GP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA/GwB,SA+GpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA/GhB,OAgHO,W;O;KA1HX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAiHP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAlHwB,SAkHpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAlHhB,OAmHO,W;O;KA7HX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAoHP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WArHwB,SAqHpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QArHhB,OAsHO,W;O;KAhIX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAuHP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAxHwB,SAwHpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAxHhB,OAyHO,W;O;KAnIX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA0HP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA3HwB,SA2HpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA3HhB,OA4HO,W;O;KAtIX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA6HP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA9HwB,SA8HpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA9HhB,OA+HO,W;O;KAzIX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAgIP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAjIwB,SAiIpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAjIhB,OAkIO,W;O;KA5IX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAmIP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WApIwB,SAoIpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QApIhB,OAqIO,W;O;KA/IX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAsIP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAvIwB,SAuIpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAvIhB,OAwIO,W;O;KAlJX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAyIP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA1IwB,SA0IpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA1IhB,OA2IO,W;O;KArJX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA4IP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA7IwB,SA6IpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA7IhB,OA8IO,W;O;KAxJX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA+IP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAhJwB,SAgJpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAhJhB,OAiJO,W;O;KA3JX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAkJP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAnJwB,SAmJpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAnJhB,OAoJO,W;O;KA9JX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAqJP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAtJwB,SAsJpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAtJhB,OAuJO,W;O;KAjKX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAwJP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAzJwB,SAyJpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAzJhB,OA0JO,W;O;KApKX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA2JP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA5JwB,SA4JpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA5JhB,OA6JO,W;O;KAvKX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA8JP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA/JwB,SA+JpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA/JhB,OAgKO,W;O;KA1KX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAiKP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAlKwB,SAkKpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAlKhB,OAmKO,W;O;KA7KX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAoKP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WArKwB,SAqKpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QArKhB,OAsKO,W;O;KAhLX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAuKP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAxKwB,SAwKpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAxKhB,OAyKO,W;O;KAnLX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA0KP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA3KwB,SA2KpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA3KhB,OA4KO,W;O;KAtLX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA6KP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA9KwB,SA8KpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA9KhB,OA+KO,W;O;KAzLX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAgLP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAjLwB,SAiLpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAjLhB,OAkLO,W;O;KA5LX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAmLP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WApLwB,SAoLpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QApLhB,OAqLO,W;O;KA/LX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAsLP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAvLwB,SAuLpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAvLhB,OAwLO,W;O;KAlMX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAyLP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA1LwB,SA0LpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA1LhB,OA2LO,W;O;KArMX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA4LP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA7LwB,SA6LpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA7LhB,OA8LO,W;O;KAxMX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA+LP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAhMwB,SAgMpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAhMhB,OAiMO,W;O;KA3MX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAkMP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAnMwB,SAmMpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAnMhB,OAoMO,W;O;KA9MX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAqMP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAtMwB,SAsMpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAtMhB,OAuMO,W;O;KAjNX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAwMP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAzMwB,SAyMpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAzMhB,OA0MO,W;O;KApNX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA2MP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA5MwB,SA4MpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA5MhB,OA6MO,W;O;KAvNX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA8MP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA/MwB,SA+MpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA/MhB,OAgNO,W;O;KA1NX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAiNP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAlNwB,SAkNpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAlNhB,OAmNO,W;O;KA7NX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAoNP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WArNwB,SAqNpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QArNhB,OAsNO,W;O;KAhOX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAuNP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAxNwB,SAwNpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAxNhB,OAyNO,W;O;KAnOX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA0NP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA3NwB,SA2NpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA3NhB,OA4NO,W;O;KAtOX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA6NP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA9NwB,SA8NpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA9NhB,OA+NO,W;O;KAzOX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAgOP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAjOwB,SAiOpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAjOhB,OAkOO,W;O;KA5OX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAmOP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WApOwB,SAoOpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QApOhB,OAqOO,W;O;KA/OX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAsOP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAvOwB,SAuOpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAvOhB,OAwOO,W;O;KAlPX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAyOP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA1OwB,SA0OpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA1OhB,OA2OO,W;O;KArPX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA4OP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA7OwB,SA6OpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA7OhB,OA8OO,W;O;KAxPX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA+OP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAhPwB,SAgPpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAhPhB,OAiPO,W;O;KA3PX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAkPP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAnPwB,SAmPpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAnPhB,OAoPO,W;O;KA9PX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAqPP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAtPwB,SAsPpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAtPhB,OAuPO,W;O;KAjQX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAwPP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAzPwB,SAyPpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAzPhB,OA0PO,W;O;KApQX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA2PP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA5PwB,SA4PpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA5PhB,OA6PO,W;O;KAvQX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA8PP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA/PwB,SA+PpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA/PhB,OAgQO,W;O;KA1QX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAiQP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAlQwB,SAkQpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAlQhB,OAmQO,W;O;KA7QX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAoQP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WArQwB,SAqQpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QArQhB,OAsQO,W;O;KAhRX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAuQP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAxQwB,SAwQpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAxQhB,OAyQO,W;O;KAnRX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA0QP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA3QwB,SA2QpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA3QhB,OA4QO,W;O;KAtRX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA6QP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA9QwB,SA8QpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA9QhB,OA+QO,W;O;KAzRX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAgRP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAjRwB,SAiRpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAjRhB,OAkRO,W;O;KA5RX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAmRP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WApRwB,SAoRpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QApRhB,OAqRO,W;O;KA/RX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAsRP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAvRwB,SAuRpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAvRhB,OAwRO,W;O;KAlSX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAyRP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA1RwB,SA0RpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA1RhB,OA2RO,W;O;KArSX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA4RP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA7RwB,SA6RpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA7RhB,OA8RO,W;O;KAxSX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA+RP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAhSwB,SAgSpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAhShB,OAiSO,W;O;KA3SX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAkSP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAnSwB,SAmSpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAnShB,OAoSO,W;O;KA9SX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAqSP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAtSwB,SAsSpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAtShB,OAuSO,W;O;KAjTX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAwSP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAzSwB,SAySpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAzShB,OA0SO,W;O;KApTX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA2SP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA5SwB,SA4SpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA5ShB,OA6SO,W;O;KAvTX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QA8SP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WA/SwB,SA+SpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA/ShB,OAgTO,W;O;KA1TX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAiTP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAlTwB,SAkTpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAlThB,OAmTO,W;O;KA7TX,C;gGAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAa,eAAa,cAAb,C;QAoTP,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WArTwB,

AAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OAAhC,C;
;MACpC,OAAO,W;K;+FAGX,yC;MAeoB,UAA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MACF,2B;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OAAhC,
C;;MACpC,OAAO,W;K;+FAGX,yC;MAeoB,UAA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MACF,2B;MAAh
B,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OAAh
C,C;;MACpC,OAAO,W;K;+FAGX,yC;MAeoB,UAA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MACF,2B;MA
AhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAnB,EAAgC,OA
AhC,C;;MACpC,OAAO,W;K;0FAGX,yB;MA1uDI,8D;MA0uDJ,gD;QAeoC,Q;QAHhC,YAtvDgB,cAAR,iBAAQ,
C;QAuvDhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAA
V,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;2FAoBA,yB;MAtvDI,8D;MAsvDJ,gD;QAeoC,Q;QAHhC,
YAlwDgB,cAAR,iBAAQ,iBAAQ,C;QAmwDhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAA
I,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;2FAoBA,yB;MAlwDI,8D;MAkwDJ,gD;QAeoC,Q;QAHhC,YA9wDgB,cAAR,iBAAQ,iBAAQ,C;QA+wDhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;2FAoBA,yB;MA9wDI,8D;MA8wDJ,gD;QAeoC,Q;QAHhC,YA1xDgB,cAAR,iBAAQ,iBAAQ,C;QA2xDhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1zDI,8D;MA0zDJ,gD;QAeoC,Q;QAHhC,YAt0DgB,cAAR,iBAAQ,iBAAQ,C;QAu0DhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAt1DI,8D;MAs1DJ,gD;QAeoC,Q;QAHhC,YAl2DgB,cAAR,iBAAQ,iBAAQ,C;QAm2DhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAl3DI,8D;MAk3DJ,gD;QAeoC,Q;QAHhC,YA93DgB,cAAR,iBAAQ,iBAAQ,C;QA+3DhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA94DI,8D;MA84DJ,gD;QAeoC,Q;QAHhC,YA15DgB,cAAR,iBAAQ,iBAAQ,C;QA25DhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA16DI,8D;MA06DJ,gD;QAeoC,Q;QAHhC,YAt7DgB,cAAR,iBAAQ,iBAAQ,C;QAu7DhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAt8DI,8D;MAs8DJ,gD;QAeoC,Q;QAHhC,YAl9DgB,cAAR,iBAAQ,iBAAQ,C;QAm9DhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAl+DI,8D;MAk+DJ,gD;QAeoC,Q;QAHhC,YA9+DgB,cAAR,iBAAQ,iBAAQ,C;QA++DhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9/DI,8D;MA8/DJ,gD;QAeoC,Q;QAHhC,YA1gEgB,cAAR,iBAAQ,iBAAQ,C;QA2gEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1hEI,8D;MA0hEJ,gD;QAeoC,Q;QAHhC,YAtiEgB,cAAR,iBAAQ,iBAAQ,C;QAuiEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtjEI,8D;MAsjEJ,gD;QAeoC,Q;QAHhC,YAlkEgB,cAAR,iBAAQ,iBAAQ,C;QAmkEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAllEI,8D;MAklEJ,gD;QAeoC,Q;QAHhC,YA9lEgB,cAAR,iBAAQ,iBAAQ,C;QA+lEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9mEI,8D;MA8mEJ,gD;QAeoC,Q;QAHhC,YA1nEgB,cAAR,iBAAQ,iBAAQ,C;QA2nEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1oEI,8D;MA0oEJ,gD;QAeoC,Q;QAHhC,YAtpEgB,cAAR,iBAAQ,iBAAQ,C;QAupEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtqEI,8D;MAsqEJ,gD;QAeoC,Q;QAHhC,YAlrEgB,cAAR,iBAAQ,iBAAQ,C;QAmrEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAlsEI,8D;MAksEJ,gD;QAeoC,Q;QAHhC,YA9sEgB,cAAR,iBAAQ,iBAAQ,C;QA+sEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9tEI,8D;MA8tEJ,gD;QAeoC,Q;QAHhC,YA1uEgB,cAAR,iBAAQ,iBAAQ,C;QA2uEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1vEI,8D;MA0vEJ,gD;QAeoC,Q;QAHhC,YAtwEgB,cAAR,iBAAQ,iBAAQ,C;QAuwEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtxEI,8D;MAsxEJ,gD;QAeoC,Q;QAHhC,YAlyEgB,cAAR,iBAAQ,iBAAQ,C;QAmyEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAlzEI,8D;MAkzEJ,gD;QAeoC,Q;QAHhC,YA9zEgB,cAAR,iBAAQ,iBAAQ,C;QA+zEhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA90EI,8D;MA80EJ,gD;QAeoC,Q;QAHhC,YA11EgB,cAAR,iBAAQ,iBAAQ,C;QA21EhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA12EI,8D;MA02EJ,gD;QAeoC,Q;QAHhC,YAt3EgB,cAAR,iBAAQ,iBAAQ,C;QAu3EhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAt4EI,8D;MAs4EJ,gD;QAeoC,Q;QAHhC,YAl5EgB,cAAR,iBAAQ,iBAAQ,C;QAm5EhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAl6EI,8D;MAk6EJ,gD;QAeoC,Q;QAHhC,YA96EgB,cAAR,iBAAQ,iBAAQ,C;QA+6EhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA97EI,8D;MA87EJ,gD;QAeoC,Q;QAHhC,YA18EgB,cAAR,iBAAQ,iBAAQ,C;QA28EhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA19EI,8D;MA09EJ,gD;QAeoC,Q;QAHhC,YAt+EgB,cAAR,iBAAQ,iBAAQ,C;QAu+EhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAt/EI,8D;MAs/EJ,gD;QAeoC,Q;QAHhC,YAlgFgB,cAAR,iBAAQ,iBAAQ,C;QAmgFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAlhFI,8D;MAkhFJ,gD;QAeoC,Q;QAHhC,YA9hFgB,cAAR,iBAAQ,iBAAQ,C;QA+hFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9iFI,8D;MA8iFJ,gD;QAeoC,Q;QAHhC,YA1jFgB,cAAR,iBAAQ,iBAAQ,C;QA2jFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1kFI,8D;MA0kFJ,gD;QAeoC,Q;QAHhC,YAtlFgB,cAAR,iBAAQ,iBAAQ,C;QAulFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtmFI,8D;MAsmFJ,gD;QAeoC,Q;QAHhC,YAlnFgB,cAAR,iBAAQ,iBAAQ,C;QAmnFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAlnFI,8D;MAknFJ,gD;QAeoC,Q;QAHhC,YA9nFgB,cAAR,iBAAQ,iBAAQ,C;QA+nFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9oFI,8D;MA8oFJ,gD;QAeoC,Q;QAHhC,YA1pFgB,cAAR,iBAAQ,iBAAQ,C;QA2pFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1qFI,8D;MA0qFJ,gD;QAeoC,Q;QAHhC,YAtrFgB,cAAR,iBAAQ,iBAAQ,C;QAurFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtsFI,8D;MAssFJ,gD;QAeoC,Q;QAHhC,YAltFgB,cAAR,iBAAQ,iBAAQ,C;QAmtFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAltFI,8D;MAktFJ,gD;QAeoC,Q;QAHhC,YA9tFgB,cAAR,iBAAQ,iBAAQ,C;QA+tFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9uFI,8D;MA8uFJ,gD;QAeoC,Q;QAHhC,YA1vFgB,cAAR,iBAAQ,iBAAQ,C;QA2vFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1wFI,8D;MA0wFJ,gD;QAeoC,Q;QAHhC,YAtxFgB,cAAR,iBAAQ,iBAAQ,C;QAuxFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtyFI,8D;MAsyFJ,gD;QAeoC,Q;QAHhC,YAlzFgB,cAAR,iBAAQ,iBAAQ,C;QAmzFhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAlzFI,8D;MAkzFJ,gD;QAeoC,Q;QAHhC,YA90FgB,cAAR,iBAAQ,iBAAQ,C;QA+0FhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA90FI,8D;MA80FJ,gD;QAeoC,Q;QAHhC,YA11FgB,cAAR,iBAAQ,iBAAQ,C;QA21FhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA12FI,8D;MA02FJ,gD;QAeoC,Q;QAHhC,YAt3FgB,cAAR,iBAAQ,iBAAQ,C;QAu3FhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAt4FI,8D;MAs4FJ,gD;QAeoC,Q;QAHhC,YAl5FgB,cAAR,iBAAQ,iBAAQ,C;QAm5FhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAl6FI,8D;MAk6FJ,gD;QAeoC,Q;QAHhC,YA96FgB,cAAR,iBAAQ,iBAAQ,C;QA+6FhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA97FI,8D;MA87FJ,gD;QAeoC,Q;QAHhC,YA18FgB,cAAR,iBAAQ,iBAAQ,C;QA28FhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA19FI,8D;MA09FJ,gD;QAeoC,Q;QAHhC,YAt+FgB,cAAR,iBAAQ,iBAAQ,C;QAu+FhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAt/FI,8D;MAs/FJ,gD;QAeoC,Q;QAHhC,YAlgGgB,cAAR,iBAAQ,iBAAQ,C;QAmgGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAlhGI,8D;MAkhGJ,gD;QAeoC,Q;QAHhC,YA9hGgB,cAAR,iBAAQ,iBAAQ,C;QA+hGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9iGI,8D;MA8iGJ,gD;QAeoC,Q;QAHhC,YA1jGgB,cAAR,iBAAQ,iBAAQ,C;QA2jGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1kGI,8D;MA0kGJ,gD;QAeoC,Q;QAHhC,YAtlGgB,cAAR,iBAAQ,iBAAQ,C;QAulGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtmGI,8D;MAsmGJ,gD;QAeoC,Q;QAHhC,YAlnGgB,cAAR,iBAAQ,iBAAQ,C;QAmnGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAlnGI,8D;MAknGJ,gD;QAeoC,Q;QAHhC,YA9nGgB,cAAR,iBAAQ,iBAAQ,C;QA+nGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9oGI,8D;MA8oGJ,gD;QAeoC,Q;QAHhC,YA1pGgB,cAAR,iBAAQ,iBAAQ,C;QA2pGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1qGI,8D;MA0qGJ,gD;QAeoC,Q;QAHhC,YAtrGgB,cAAR,iBAAQ,iBAAQ,C;QAurGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtsGI,8D;MAssGJ,gD;QAeoC,Q;QAHhC,YAltGgB,cAAR,iBAAQ,iBAAQ,C;QAmtGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAltGI,8D;MAktGJ,gD;QAeoC,Q;QAHhC,YA9tGgB,cAAR,iBAAQ,iBAAQ,C;QA+tGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9uGI,8D;MA8uGJ,gD;QAeoC,Q;QAHhC,YA1vGgB,cAAR,iBAAQ,iBAAQ,C;QA2vGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1wGI,8D;MA0wGJ,gD;QAeoC,Q;QAHhC,YAtxGgB,cAAR,iBAAQ,iBAAQ,C;QAuxGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtyGI,8D;MAsyGJ,gD;QAeoC,Q;QAHhC,YAlzGgB,cAAR,iBAAQ,iBAAQ,C;QAmzGhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAlzGI,8D;MAkzGJ,gD;QAeoC,Q;QAHhC,YA90GgB,cAAR,iBAAQ,iBAAQ,C;QA+0GhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA90GI,8D;MA80GJ,gD;QAeoC,Q;QAHhC,YA11GgB,cAAR,iBAAQ,iBAAQ,C;QA21GhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA12GI,8D;MA02GJ,gD;QAeoC,Q;QAHhC,YAt3GgB,cAAR,iBAAQ,iBAAQ,C;QAu3GhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAt4GI,8D;MAs4GJ,gD;QAeoC,Q;QAHhC,YAl5GgB,cAAR,iBAAQ,iBAAQ,C;QAm5GhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAl6GI,8D;MAk6GJ,gD;QAeoC,Q;QAHhC,YA96GgB,cAAR,iBAAQ,iBAAQ,C;QA+6GhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA97GI,8D;MA87GJ,gD;QAeoC,Q;QAHhC,YA18GgB,cAAR,iBAAQ,iBAAQ,C;QA28GhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA19GI,8D;MA09GJ,gD;QAeoC,Q;QAHhC,YAt+GgB,cAAR,iBAAQ,iBAAQ,C;QAu+GhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAt/GI,8D;MAs/GJ,gD;QAeoC,Q;QAHhC,YAlgHgB,cAAR,iBAAQ,iBAAQ,C;QAmgHhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAlhHI,8D;MAkhHJ,gD;QAeoC,Q;QAHhC,YA9hHgB,cAAR,iBAAQ,iBAAQ,C;QA+hHhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9iHI,8D;MA8iHJ,gD;QAeoC,Q;QAHhC,YA1jHgB,cAAR,iBAAQ,iBAAQ,C;QA2jHhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1kHI,8D;MA0kHJ,gD;QAeoC,Q;QAHhC,YAtlHgB,cAAR,iBAAQ,iBAAQ,C;QAulHhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtmHI,8D;MAsmHJ,gD;QAeoC,Q;QAHhC,YAlnHgB,cAAR,iBAAQ,iBAAQ,C;QAmnHhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAlnHI,8D;MAknHJ,gD;QAeoC,Q;QAHhC,YA9nHgB,cAAR,iBAAQ,iBAAQ,C;QA+nHhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9oHI,8D;MA8oHJ,gD;QAeoC,Q;QAHhC,YA1pHgB,cAAR,iBAAQ,iBAAQ,C;QA2pHhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA1qHI,8D;MA0qHJ,gD;QAeoC,Q;QAHhC,YAtrHgB,cAAR,iBAAQ,iBAAQ,C;QAurHhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAtsHI,8D;MAssHJ,gD;QAeoC,Q;QAHhC,YAltHgB,cAAR,iBAAQ,iBAAQ,C;QAmtHhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MAltHI,8D;MAktHJ,gD;QAeoC,Q;QAHhC,YA9tHgB,cAAR,iBAAQ,iBAAQ,C;QA+tHhB,kBAAkB,O;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAjBX,C;yFAoBA,yB;MA9uHI,8D;MA8uHJ,gD;QAeoC,Q;QAHhC,YA1vHgB,cAAR,iBAAQ,iBAAQ,C;QA2vHhB,kBA

W,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QApFP,yB;O;KANJ,C;mFASA,yB;MAjhEI,8D;MAihEJ,sC;QAMW,sB;;UAuFP,IAAI,mBAAJ,C;YAAe,qBAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBAhnEgB,cAAR,iBAAAQ,C;UAinEhB,IAAI,cAAa,CAAjB,B,C;YAAoB,qBAAO,O;YAAP,uB;;UACpB,eA3FmB,QA2FJ,CAAS,OAAT,C;UACf,aAAU,CAAV,OAAa,SAAb,M;YACI,QAAQ,sBAAK,CAAL,C;YACR,QA9Fe,QA8FP,CAAS,CAAT,C;YACR,IAAI,2BAAW,CAAX,KAAJ,C;cACI,UAAU,C;cACV,WAAW,C;;;UAGnB,qBAAO,O;;;QApGP,yB;O;KANJ,C;+FASA,yB;MAljEI,8D;MAkjEJ,sC;QASI,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,cAAc,sBAAK,CAAL,C;QACd,gBA7jEgB,cA6jEA,SA7jEA,cA6jEA,QAAR,iBAAAQ,C;QAAhB,IAAI,cAAa,CAAjB,B,C;UAAoB,qBAAO,O;UAAP,uB;;QACpB,eAAO,O;QAAP,yB;;MAGJ,yB;MAAA,yB;;IAGJ,yB;IAAA,yB;;EAGJ,yB;EAAA,yB;;AAGJ,yB;AAAA,yB;;AAGA,yB;AAAA,yB;;AAGA,yB;AAAA,yB;;AAGA

BilMd,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA90EI,8D;MA80EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA51EG,cAAR,iBAAQ,C;QA41EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA71EI,8D;MA61EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA32EG,cAAR,iBAAQ,C;QA22EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA52EI,8D;MA42EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA13EG,cAAR,iBAAQ,C;QA03EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA33EI,8D;MA23EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAz4EG,cAAR,iBAAQ,C;QAy4EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA14EI,8D;MA04EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAx5EG,cAAR,iBAAQ,C;QAw5EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAz5EI,8D;MAy5EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAv6EG,cAAR,iBAAQ,C;QAu6EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAx6EI,8D;MAw6EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAt7EG,cAAR,iBAAQ,C;QAs7EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAv7EI,8D;MAu7EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAr8EG,cAAR,iBAAQ,C;QAq8EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAt8EI,8D;MAs8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAp9EG,cAAR,iBAAQ,C;QAo9EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAr9EI,8D;MAq9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAn+EG,cAAR,iBAAQ,C;QAm+EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAp+EI,8D;MAo+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAl/EG,cAAR,iBAAQ,C;QAk/EhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAn/EI,8D;MAm/EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjgFG,cAAR,iBAAQ,C;QAigFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAl/EI,8D;MAk/EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhgFG,cAAR,iBAAQ,C;QAggFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAj/EI,8D;MAi/EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/fG,cAAR,iBAAQ,C;QA+fhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAh/EI,8D;MAg/EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9fG,cAAR,iBAAQ,C;QA8fhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA/+EI,8D;MA++EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7fG,cAAR,iBAAQ,C;QA6fhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA9+EI,8D;MA8+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA5fG,cAAR,iBAAQ,C;QA4fhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA7+EI,8D;MA6+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA3fG,cAAR,iBAAQ,C;QA2fhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA5+EI,8D;MA4+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA1fG,cAAR,iBAAQ,C;QA0fhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA3+EI,8D;MA2+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAzfG,cAAR,iBAAQ,C;QAyfhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA1+EI,8D;MA0+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxfG,cAAR,iBAAQ,C;QAwfhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAz+EI,8D;MAy+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAvfG,cAAR,iBAAQ,C;QAufhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAx+EI,8D;MAw+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAtfG,cAAR,iBAAQ,C;QAsfhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAv+EI,8D;MAu+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OArfG,cAAR,iBAAQ,C;QAqfhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAt+EI,8D;MAs+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApfG,cAAR,iBAAQ,C;QAofhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAr+EI,8D;MAq+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAnfG,cAAR,iBAAQ,C;QAmfhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAp+EI,8D;MAo+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAlfG,cAAR,iBAAQ,C;QAkfhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAn+EI,8D;MAm+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjfG,cAAR,iBAAQ,C;QAifhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAl+EI,8D;MAk+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhfG,cAAR,iBAAQ,C;QAgfhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAj+EI,8D;MAi+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/eG,cAAR,iBAAQ,C;QA+ehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAh+EI,8D;MAg+EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9eG,cAAR,iBAAQ,C;QA8ehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA/9EI,8D;MA+9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7eG,cAAR,iBAAQ,C;QA6ehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA99EI,8D;MA89EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA5eG,cAAR,iBAAQ,C;QA4ehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA79EI,8D;MA69EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA3eG,cAAR,iBAAQ,C;QA2ehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA59EI,8D;MA49EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA1eG,cAAR,iBAAQ,C;QA0ehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA39EI,8D;MA29EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAzeG,cAAR,iBAAQ,C;QAyehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA19EI,8D;MA09EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxeG,cAAR,iBAAQ,C;QAwehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAz9EI,8D;MAy9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAveG,cAAR,iBAAQ,C;QAuehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAx9EI,8D;MAw9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAteG,cAAR,iBAAQ,C;QAsehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAv9EI,8D;MAu9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAreG,cAAR,iBAAQ,C;QAqehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAt9EI,8D;MAs9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApeG,cAAR,iBAAQ,C;QAoehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAr9EI,8D;MAq9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAneG,cAAR,iBAAQ,C;QAmehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAp9EI,8D;MAo9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAleG,cAAR,iBAAQ,C;QAkehB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAn9EI,8D;MAm9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjeG,cAAR,iBAAQ,C;QAieX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAl9EI,8D;MAk9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAheG,cAAR,iBAAQ,C;QAgeX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAj9EI,8D;MAi9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/dG,cAAR,iBAAQ,C;QA+dX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAh9EI,8D;MAg9EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9dG,cAAR,iBAAQ,C;QA8dX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA/8EI,8D;MA+8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7dG,cAAR,iBAAQ,C;QA6dX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA98EI,8D;MA88EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA5dG,cAAR,iBAAQ,C;QA4dX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA78EI,8D;MA68EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA3dG,cAAR,iBAAQ,C;QA2dX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA58EI,8D;MA48EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA1dG,cAAR,iBAAQ,C;QA0dX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA38EI,8D;MA28EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAzdG,cAAR,iBAAQ,C;QAydX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA18EI,8D;MA08EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxdG,cAAR,iBAAQ,C;QAwdX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAz8EI,8D;MAy8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAvdG,cAAR,iBAAQ,C;QAudX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAx8EI,8D;MAw8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAtdG,cAAR,iBAAQ,C;QAsdX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAv8EI,8D;MAu8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OArdG,cAAR,iBAAQ,C;QAqdX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAt8EI,8D;MAs8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApdG,cAAR,iBAAQ,C;QAodX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAr8EI,8D;MAq8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAndG,cAAR,iBAAQ,C;QAmdX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAp8EI,8D;MAo8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAldG,cAAR,iBAAQ,C;QAkdX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAn8EI,8D;MAm8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjdG,cAAR,iBAAQ,C;QAidX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAl8EI,8D;MAk8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhdG,cAAR,iBAAQ,C;QAgdX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAj8EI,8D;MAi8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/cG,cAAR,iBAAQ,C;QA+cX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAh8EI,8D;MAg8EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9cG,cAAR,iBAAQ,C;QA8cX,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA/7EI,8D;MA+7EJ,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,

I,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAroFG,cAAR,iBAAQ,C;QAqoFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;2FAqBA,yB;MAAA,sE;MAtqFI,8D;MAsqFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAprFG,cAAR,iBAAQ,C;QAorFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;0FAuBA,yB;MAAA,sE;MArrFI,8D;MAqrFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAnsFG,cAAR,iBAAQ,C;QAmsFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAtsFI,8D;MAssFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAptFG,cAAR,iBAAQ,C;QAotFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAvtFI,8D;MAutFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OArtFG,cAAR,iBAAQ,C;QAqtFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAxtFI,8D;MAwtFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAttFG,cAAR,iBAAQ,C;QAstFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAztFI,8D;MAytFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAvtFG,cAAR,iBAAQ,C;QAutFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA1tFI,8D;MA0tFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxtFG,cAAR,iBAAQ,C;QAwtFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA3tFI,8D;MA2tFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAztFG,cAAR,iBAAQ,C;QAytFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA5tFI,8D;MA4tFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA1tFG,cAAR,iBAAQ,C;QA0tFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA7tFI,8D;MA6tFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA3tFG,cAAR,iBAAQ,C;QA2tFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA9tFI,8D;MA8tFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA5tFG,cAAR,iBAAQ,C;QA4tFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA/tFI,8D;MA+tFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7tFG,cAAR,iBAAQ,C;QA6tFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAhuFI,8D;MAguFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9tFG,cAAR,iBAAQ,C;QA8tFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAjuFI,8D;MAiuFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/tFG,cAAR,iBAAQ,C;QA+tFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAluFI,8D;MAkuFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhuFG,cAAR,iBAAQ,C;QAguFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAnuFI,8D;MAmuFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjuFG,cAAR,iBAAQ,C;QAiuFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MApuFI,8D;MAouFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAluFG,cAAR,iBAAQ,C;QAkuFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAruFI,8D;MAquFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAnuFG,cAAR,iBAAQ,C;QAmuFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAtuFI,8D;MAsuFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApuFG,cAAR,iBAAQ,C;QAouFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAvuFI,8D;MAuuFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAruFG,cAAR,iBAAQ,C;QAquFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAxuFI,8D;MAwuFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAtuFG,cAAR,iBAAQ,C;QAsuFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAzuFI,8D;MAyuFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAvuFG,cAAR,iBAAQ,C;QAuuFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA1uFI,8D;MA0uFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxuFG,cAAR,iBAAQ,C;QAwuFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA3uFI,8D;MA2uFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAzuFG,cAAR,iBAAQ,C;QAyuFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA5uFI,8D;MA4uFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA1uFG,cAAR,iBAAQ,C;QA0uFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA7uFI,8D;MA6uFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA3uFG,cAAR,iBAAQ,C;QA2uFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA9uFI,8D;MA8uFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA5uFG,cAAR,iBAAQ,C;QA4uFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA/uFI,8D;MA+uFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7uFG,cAAR,iBAAQ,C;QA6uFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAhvFI,8D;MAgvFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9uFG,cAAR,iBAAQ,C;QA8uFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAjvFI,8D;MAivFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/uFG,cAAR,iBAAQ,C;QA+uFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAlvFI,8D;MAkvFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhvFG,cAAR,iBAAQ,C;QAgvFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAnvFI,8D;MAmvFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjvFG,cAAR,iBAAQ,C;QAivFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MApvFI,8D;MAovFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAlvFG,cAAR,iBAAQ,C;QAkvFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MArvFI,8D;MAqvFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAnvFG,cAAR,iBAAQ,C;QAmvFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAtvFI,8D;MAsvFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApvFG,cAAR,iBAAQ,C;QAovFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAvvFI,8D;MAuvFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OArvFG,cAAR,iBAAQ,C;QAqvFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAxvFI,8D;MAwvFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAtvFG,cAAR,iBAAQ,C;QAsvFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAzvFI,8D;MAyvFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAvvFG,cAAR,iBAAQ,C;QAuvFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA1vFI,8D;MA0vFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxvFG,cAAR,iBAAQ,C;QAwvFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA3vFI,8D;MA2vFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAzvFG,cAAR,iBAAQ,C;QAyvFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA5vFI,8D;MA4vFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA1vFG,cAAR,iBAAQ,C;QA0vFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA7vFI,8D;MA6vFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA3vFG,cAAR,iBAAQ,C;QA2vFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA9vFI,8D;MA8vFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA5vFG,cAAR,iBAAQ,C;QA4vFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MA/vFI,8D;MA+vFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7vFG,cAAR,iBAAQ,C;QA6vFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAhwFI,8D;MAgwFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9vFG,cAAR,iBAAQ,C;QA8vFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAjwFI,8D;MAiwFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/vFG,cAAR,iBAAQ,C;QA+vFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAlwFI,8D;MAkwFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhwFG,cAAR,iBAAQ,C;QAgwFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAnwFI,8D;MAmwFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjwFG,cAAR,iBAAQ,C;QAiwFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MApwFI,8D;MAowFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAlwFG,cAAR,iBAAQ,C;QAkwFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MArwFI,8D;MAqwFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAnwFG,cAAR,iBAAQ,C;QAmwFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAtwFI,8D;MAswFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApwFG,cAAR,iBAAQ,C;QAowFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;2FAuBA,yB;MAAA,sE;MAvwFI,8D;MAuwFJ,kD;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OArwFG,cAAR,iBAAQ,C;QAqwFhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,kB,CAAqB,CAAX,EAAe,CAAf,kC,CAAkB,CAAlB,C;YACI,WAA

AAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,iD;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAx8FG,gBAAR,iBAAAQ,C;MAw8FhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,iD;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA/8FG,gBAAR,iBAAAQ,C;MA+8FhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA9gGI,8D;MA8gGJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA7jGG,gBAAR,iBAAAQ,C;UA6jGH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA9iGI,8D;MA8iGJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA7lGG,gBAAR,iBAAAQ,C;UA6lGH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA9kGI,8D;MA8kGJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA7nGG,gBAAR,iBAAAQ,C;UA6nGH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA9mGI,8D;MA8mGJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA7pGG,gBAAR,iBAAAQ,C;UA6pGH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA9oGI,8D;MA8oGJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA7rGG,gBAAR,iBAAAQ,C;UA6rGH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA9qGI,8D;MA8qGJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA7tGG,gBAAR,iBAAAQ,C;UA6tGH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA9sGI,8D;MA8sGJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA7vGG,gBAAR,iBAAAQ,C;UA6vGH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA9uGI,8D;MA8uGJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA7xGG,gBAAR,iBAAAQ,C;UA6xGH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA9wGI,8D;MA8wGJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA7zGG,gBAAR,iBAAAQ,C;UA6zGH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA9yGI,8D;MA8yGJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA72GG,gBAAR,iBAAAQ,C;UA62GH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA91GI,8D;MA81GJ,gJ,sC;QAMW,W,sB;;UAuCP,IAAI,mBAAJ,C;YAAe,eAAO,I;YAAP,uB;;UACf,cAAc,sBAAK,CAAL,C;UACd,gBA74GG,gBAAR,iBAAAQ,C;UA64GH,cAAc,iBAAd,iB;UACI,QAAQ,sBAAK,CAAL,C;UACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;YAAoC,MAAM,C;;UAE9C,OAAO,G;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;IAGX,2B;MAKI,OAAO,uB;K;mFAGX,yB;MA95GI,8D;MA85

FAuBA,yB;MAAA,sE;MAprGI,8D;MpB/jHJ,iB;MoBmvNA,sC;QAgBiB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM
,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApsGG,cAAR,iBAAQ,C;QAosGhB,aAAU,CAAV,iB;
UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB7vNG,MAAO,KoB6vNO,QpB7vNP,EoB6vNiB,CpB7v
NjB,C;;QoB+vNd,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAnsGI,8D;MpBvkHJ,iB;MoB0wNA,sC;QAg
BiB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAntGG,c
AAR,iBAAQ,C;QAmtGhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBpxNG,M
AO,KoBoxNO,QpBpxNP,EoBoxNiB,CpBpxNjB,C;;QoBsxNd,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;
MAlvGI,8D;MpB1jHJ,iB;MoB4yNA,sC;QAgBiB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SA
AS,sBAAK,CAAL,CAAT,C;QACF,OAlwGG,cAAR,iBAAQ,C;QAkwGhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,
sBAAK,CAAL,CAAT,C;UACR,WpBtzNG,MAAO,KoBszNO,QpBtzNP,EoBszNiB,CpBtzNjB,C;;QoBwzNd,OAA
O,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAjwGI,8D;MpBllHJ,iB;MoBm0NA,sC;QAgBiB,Q;QAFb,IAAI,mB
AAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjxGG,cAAR,iBAAQ,C;QAixGh
B,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB70NG,MAAO,KoB60NO,QpB70
NP,EoB60NiB,CpB70NjB,C;;QoB+0Nd,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MAhxGI,8D;MpB1kHJ,
iB;MoB01NA,sC;QAgBiB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CA
AT,C;QACF,OAhyGG,cAAR,iBAAQ,C;QAgyGhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;
UACR,WpBp2NG,MAAO,KoBo2NO,QpBp2NP,EoBo2NiB,CpBp2NjB,C;;QoBs2Nd,OAAO,Q;O;KApBX,C;m
FAuBA,yB;MAAA,sE;MA/xGI,8D;MpBjlHJ,iB;MoBi3NA,sC;QAgBiB,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,
6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/yGG,cAAR,iBAAQ,C;QA+yGhB,aAAU,CAAV,iB;
UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB33NG,MAAO,KoB23NO,QpB33NP,EoB23NiB,CpB33
NjB,C;;QoB63Nd,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA90GI,8D;MpBrlHJ,iB;MoB44NA,sC;QAciB,
Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA51GG,cAAR,
iBAAQ,C;QA41GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA71GI,8D;MpBvlHJ,iB;MoB85NA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA32GG,cAAR,
iBAAQ,C;QA22GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA52GI,8D;MpBzlHJ,iB;MoBg8NA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA13GG,cAAR,
iBAAQ,C;QA03GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;mFAuBA,yB;MAAA,sE;MA33GI,8D;MpB3lHJ,iB;MoBk+NA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAz4GG,cAAR,
iBAAQ,C;QAy4GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KApBX,C;8FAuBA,yB;MAAA,sE;MA14GI,8D;MpB7lHJ,iB;MoBo/NA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,MAAM,6B;QACrB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAx5GG,cAAR,
iBAAQ,C;QAw5GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAz5GI,8D;MpB/lHJ,iB;MoBsgOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAv6GG,cAAR,
iBAAQ,C;QAu6GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAx6GI,8D;MpBjmHJ,iB;MoBwhOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAt7GG,cAAR,
iBAAQ,C;QAs7GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAv7GI,8D;MpBnmHJ,iB;MoB0iOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAr8GG,cAAR,
iBAAQ,C;QAq8GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAt8GI,8D;MpBrmHJ,iB;MoB4jOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAp9GG,cAAR,
iBAAQ,C;QAo9GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAr9GI,8D;MpBvmHJ,iB;MoB8kOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAn+GG,cAAR,
iBAAQ,C;QAm+GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAp+GI,8D;MpBzmHJ,iB;MoBgmOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAl/GG,cAAR,
iBAAQ,C;QAk/GhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAn/GI,8D;MpB3mHJ,iB;MoBsnOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjgHG,cAAR,
iBAAQ,C;QAigHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAl/GI,8D;MpB7mHJ,iB;MoB4oOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhhHG,cAAR,
iBAAQ,C;QAghHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAi/GI,8D;MpB/mHJ,iB;MoBkpOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/hHG,cAAR,
iBAAQ,C;QA+hHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAh/GI,8D;MpBjnHJ,iB;MoBwqOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9iHG,cAAR,
iBAAQ,C;QA8iHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA//GI,8D;MpBnnHJ,iB;MoB8rOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7jHG,cAAR,
iBAAQ,C;QA6jHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA9/GI,8D;MpBrnHJ,iB;MoBosOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA5kHG,cAAR,
iBAAQ,C;QA4kHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA7gHI,8D;MpBvnHJ,iB;MoB0tOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA3lHG,cAAR,
iBAAQ,C;QA2lHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA5hHI,8D;MpBznHJ,iB;MoBguOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA1mHG,cAAR,
iBAAQ,C;QA0mHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA3iHI,8D;MpB3nHJ,iB;MoBsvOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAznHG,cAAR,
iBAAQ,C;QAynHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA1jHI,8D;MpB7nHJ,iB;MoB4wOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxoHG,cAAR,
iBAAQ,C;QAwoHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAzkHI,8D;MpB/nHJ,iB;MoBkyOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAvpHG,cAAR,
iBAAQ,C;QAupHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAxlHI,8D;MpBjoHJ,iB;MoB0zOA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAtqHG,cAAR,
iBAAQ,C;QAsqHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAvmHI,8D;MpBnoHJ,iB;MoB40OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OArrHG,cAAR,
iBAAQ,C;QAqrHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAtnHI,8D;MpBroHJ,iB;MoB81OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApsHG,cAAR,
iBAAQ,C;QAosHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAroHI,8D;MpBvoHJ,iB;MoBg3OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAntHG,cAAR,
iBAAQ,C;QAmtHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MApqHI,8D;MpBzoHJ,iB;MoBk4OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAluHG,cAAR,
iBAAQ,C;QAkuHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAnrHI,8D;MpB3oHJ,iB;MoBo5OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjvHG,cAAR,
iBAAQ,C;QAivHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAlsHI,8D;MpB7oHJ,iB;MoBs6OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhwHG,cAAR,
iBAAQ,C;QAgwHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAjtHI,8D;MpB/oHJ,iB;MoBw7OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/wHG,cAAR,
iBAAQ,C;QA+wHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAhuHI,8D;MpBjpHJ,iB;MoB08OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9xHG,cAAR,
iBAAQ,C;QA8xHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA/uHI,8D;MpBnpHJ,iB;MoB49OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7yHG,cAAR,
iBAAQ,C;QA6yHhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA9vHI,8D;MpBrpHJ,iB;MoB8+OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA50HG,cAAR,
iBAAQ,C;QA40HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA7wHI,8D;MpBvpHJ,iB;MoBs/OA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA31HG,cAAR,
iBAAQ,C;QA21HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA5xHI,8D;MpBzpHJ,iB;MoBggPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA12HG,cAAR,
iBAAQ,C;QA02HhB,aAAU,CAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA3yHI,8D;MpB3pHJ,iB;MoB0gPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAz3HG,cAAR,
iBAAQ,C;QAy3HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA1zHI,8D;MpB7pHJ,iB;MoBohPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAx4HG,cAAR,
iBAAQ,C;QAw4HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAz0HI,8D;MpB/pHJ,iB;MoB2hPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAv5HG,cAAR,
iBAAQ,C;QAu5HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAx1HI,8D;MpBjqHJ,iB;MoBkiPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAt6HG,cAAR,
iBAAQ,C;QAs6HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAv2HI,8D;MpBnqHJ,iB;MoByiPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAr7HG,cAAR,
iBAAQ,C;QAq7HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAt3HI,8D;MpBrqHJ,iB;MoBgjPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAp8HG,cAAR,
iBAAQ,C;QAo8HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAr4HI,8D;MpBvqHJ,iB;MoBujPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAn9HG,cAAR,
iBAAQ,C;QAm9HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAp5HI,8D;MpBzqHJ,iB;MoBqkPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAl+HG,cAAR,
iBAAQ,C;QAk+HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAn6HI,8D;MpB3qHJ,iB;MoBmlPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAj/HG,cAAR,
iBAAQ,C;QAi/HhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAl7HI,8D;MpB7qHJ,iB;MoBimPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhgIG,cAAR,
iBAAQ,C;QAggIhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAi7HI,8D;MpB/qHJ,iB;MoBkpPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/gIG,cAAR,
iBAAQ,C;QA+gIhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAh7HI,8D;MpBjrHJ,iB;MoBwsPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9hIG,cAAR,
iBAAQ,C;QA8hIhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA/7HI,8D;MpBnrHJ,iB;MoB8vPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7jIG,cAAR,
iBAAQ,C;QA6jIhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA98HI,8D;MpBvrHJ,iB;MoBqyPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA5kIG,cAAR,
iBAAQ,C;QA4kIhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA38HI,8D;MpBzrHJ,iB;MoB+zPA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA3lIG,cAAR,
iBAAQ,C;QA2lIhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MA18HI,8D;MpB3rHJ,iB;MoBo1PA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxmIG,cAAR,
iBAAQ,C;QAwmIhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAz9HI,8D;MpB7rHJ,iB;MoBy2PA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAroIG,cAAR,
iBAAQ,C;QAqoIhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAt+HI,8D;MpBvsHJ,iB;MoBi4PA,sC;QAci
B,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAlqIG,cAAR,
iBAAQ,C;QAkqIhB,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;
YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAAA,sE;MAn/H

BhkOG,MAAO,KoBgkOO,QpBhkOP,EoBgkOiB,CpBhkOjB,C;;QoBkkOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA3gHI,8D;MpBlkHJ,iB;MoB6kOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAzhHG,cAAR,iBAAQ,CAAQ,C;QAyhHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBrlOG,MAAO,KoBqlOO,QpBrlOP,EoBqlOiB,CpBrlOjB,C;;QoBulOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAxhHI,8D;MpB1kHJ,iB;MoBkmOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAtiHG,cAAR,iBAAQ,CAAQ,C;QAsiHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB1mOG,MAAO,KoB0mOO,QpB1mOP,EoB0mOiB,CpB1mOjB,C;;QoB4mOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA7iHI,8D;MpB/lHJ,iB;MoBunOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA3jHG,cAAR,iBAAQ,CAAQ,C;QA2jHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB/nOG,MAAO,KoB+nOO,QpB/nOP,EoB+nOiB,CpB/nOjB,C;;QoBioOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAlkHI,8D;MpBpnHJ,iB;MoB4oOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhlHG,cAAR,iBAAQ,CAAQ,C;QAglHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBppOG,MAAO,KoBopOO,QpBppOP,EoBopOiB,CpBppOjB,C;;QoBspOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAvlHI,8D;MpBzoHJ,iB;MoBiqOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OArmHG,cAAR,iBAAQ,CAAQ,C;QAqmHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBzqOG,MAAO,KoByqOO,QpBzqOP,EoByqOiB,CpBzqOjB,C;;QoB2qOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA5mHI,8D;MpB9pHJ,iB;MoBsrOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA1nHG,cAAR,iBAAQ,CAAQ,C;QA0nHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB9rOG,MAAO,KoB8rOO,QpB9rOP,EoB8rOiB,CpB9rOjB,C;;QoBgsOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAjoHI,8D;MpBnrHJ,iB;MoB2sOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/oHG,cAAR,iBAAQ,CAAQ,C;QA+oHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBntOG,MAAO,KoBmtOO,QpBntOP,EoBmtOiB,CpBntOjB,C;;QoBqtOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAtpHI,8D;MpBxsHJ,iB;MoBguOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApqHG,cAAR,iBAAQ,CAAQ,C;QAoqHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBxuOG,MAAO,KoBwuOO,QpBxuOP,EoBwuOiB,CpBxuOjB,C;;QoB0uOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA3qHI,8D;MpB7tHJ,iB;MoBqvOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAzrHG,cAAR,iBAAQ,CAAQ,C;QAyrHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB7vOG,MAAO,KoB6vOO,QpB7vOP,EoB6vOiB,CpB7vOjB,C;;QoB+vOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAhsHI,8D;MpBlvHJ,iB;MoB0wOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9sHG,cAAR,iBAAQ,CAAQ,C;QA8sHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBlxOG,MAAO,KoBkxOO,QpBlxOP,EoBkxOiB,CpBlxOjB,C;;QoBoxOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MArtHI,8D;MpBvwHJ,iB;MoB+xOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAnuHG,cAAR,iBAAQ,CAAQ,C;QAmuHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBvyOG,MAAO,KoBuyOO,QpBvyOP,EoBuyOiB,CpBvyOjB,C;;QoByyOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA1uHI,8D;MpB5xHJ,iB;MoBozOA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxvHG,cAAR,iBAAQ,CAAQ,C;QAwvHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB5zOG,MAAO,KoB4zOO,QpB5zOP,EoB4zOiB,CpB5zOjB,C;;QoB8zOd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA/vHI,8D;MpBjzHJ,iB;MoBy0OA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7wHG,cAAR,iBAAQ,CAAQ,C;QA6wHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBj1OG,MAAO,KoBi1OO,QpBj1OP,EoBi1OiB,CpBj1OjB,C;;QoBm1Od,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MApxHI,8D;MpBt0HJ,iB;MoB81OA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAlyHG,cAAR,iBAAQ,CAAQ,C;QAkyHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBt2OG,MAAO,KoBs2OO,QpBt2OP,EoBs2OiB,CpBt2OjB,C;;QoBw2Od,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAzyHI,8D;MpB31HJ,iB;MoBm3OA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAvzHG,cAAR,iBAAQ,CAAQ,C;QAuzHb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB33OG,MAAO,KoB23OO,QpB33OP,EoB23OiB,CpB33OjB,C;;QoB63Od,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA9zHI,8D;MpBh3HJ,iB;MoBw4OA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA50HG,cAAR,iBAAQ,CAAQ,C;QA40Hb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBh5OG,MAAO,KoBg5OO,QpBh5OP,EoBg5OiB,CpBh5OjB,C;;QoBk5Od,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAn1HI,8D;MpBr4HJ,iB;MoB65OA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAj2HG,cAAR,iBAAQ,CAAQ,C;QAi2Hb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBr6OG,MAAO,KoBq6OO,QpBr6OP,EoBq6OiB,CpBr6OjB,C;;QoBu6Od,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAx2HI,8D;MpB15HJ,iB;MoBk7OA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAt3HG,cAAR,iBAAQ,CAAQ,C;QAs3Hb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB17OG,MAAO,KoB07OO,QpB17OP,EoB07OiB,CpB17OjB,C;;QoB47Od,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA73HI,8D;MpB/6HJ,iB;MoBu8OA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA34HG,cAAR,iBAAQ,CAAQ,C;QA24Hb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB/8OG,MAAO,KoB+8OO,QpB/8OP,EoB+8OiB,CpB/8OjB,C;;QoBi9Od,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAl5HI,8D;MpBp8HJ,iB;MoB4+OA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAh6HG,cAAR,iBAAQ,CAAQ,C;QAg6Hb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBp/OG,MAAO,KoBo/OO,QpBp/OP,EoBo/OiB,CpBp/OjB,C;;QoBs/Od,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAv6HI,8D;MpBz+HJ,iB;MoBigPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAr7HG,cAAR,iBAAQ,CAAQ,C;QAq7Hb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBzgPG,MAAO,KoBygPO,QpBzgPP,EoBygPiB,CpBzgPjB,C;;QoB2gPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA57HI,8D;MpB9/HJ,iB;MoBshPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA18HG,cAAR,iBAAQ,CAAQ,C;QA08Hb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB9hPG,MAAO,KoB8hPO,QpB9hPP,EoB8hPiB,CpB9hPjB,C;;QoBgiPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAj9HI,8D;MpBngIJ,iB;MoB2iPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/9HG,cAAR,iBAAQ,CAAQ,C;QA+9Hb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBnjPG,MAAO,KoBmjPO,QpBnjPP,EoBmjPiB,CpBnjPjB,C;;QoBqjPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAt+HI,8D;MpBxhIJ,iB;MoBgkPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApgIG,cAAR,iBAAQ,CAAQ,C;QAogIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBxkPG,MAAO,KoBwkPO,QpBxkPP,EoBwkPiB,CpBxkPjB,C;;QoB0kPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA3gII,8D;MpB7iIJ,iB;MoBqlPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAzhIG,cAAR,iBAAQ,CAAQ,C;QAyhIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB7lPG,MAAO,KoB6lPO,QpB7lPP,EoB6lPiB,CpB7lPjB,C;;QoB+lPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAhiII,8D;MpBlkIJ,iB;MoB0mPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9iIG,cAAR,iBAAQ,CAAQ,C;QA8iIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBlnPG,MAAO,KoBknPO,QpBlnPP,EoBknPiB,CpBlnPjB,C;;QoBonPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MArjII,8D;MpBvlIJ,iB;MoB+nPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAnkIG,cAAR,iBAAQ,CAAQ,C;QAmkIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBvoPG,MAAO,KoBuoPO,QpBvoPP,EoBuoPiB,CpBvoPjB,C;;QoByoPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA1kII,8D;MpB5mIJ,iB;MoBoqPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxlIG,cAAR,iBAAQ,CAAQ,C;QAwlIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB5pPG,MAAO,KoB4pPO,QpB5pPP,EoB4pPiB,CpB5pPjB,C;;QoB8pPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAvlII,8D;MpBznIJ,iB;MoBirPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OArmIG,cAAR,iBAAQ,CAAQ,C;QAqmIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBzqPG,MAAO,KoByqPO,QpBzqPP,EoByqPiB,CpBzqPjB,C;;QoB2qPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MApmII,8D;MpBtoIJ,iB;MoB8rPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAlnIG,cAAR,iBAAQ,CAAQ,C;QAknIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBtrPG,MAAO,KoBsrPO,QpBtrPP,EoBsrPiB,CpBtrPjB,C;;QoBwrPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAjnII,8D;MpBnpIJ,iB;MoB2sPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/nIG,cAAR,iBAAQ,CAAQ,C;QA+nIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBntPG,MAAO,KoBmtPO,QpBntPP,EoBmtPiB,CpBntPjB,C;;QoBqtPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAhoII,8D;MpBhqIJ,iB;MoBwuPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA5oIG,cAAR,iBAAQ,CAAQ,C;QA4oIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBhvPG,MAAO,KoBgvPO,QpBhvPP,EoBgvPiB,CpBhvPjB,C;;QoBkvPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA7oII,8D;MpB7rIJ,iB;MoBqwPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAzpIG,cAAR,iBAAQ,CAAQ,C;QAypIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB7wPG,MAAO,KoB6wPO,QpB7wPP,EoB6wPiB,CpB7wPjB,C;;QoB+wPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA1pII,8D;MpB1sIJ,iB;MoBkxPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAtqIG,cAAR,iBAAQ,CAAQ,C;QAsqIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB1xPG,MAAO,KoB0xPO,QpB1xPP,EoB0xPiB,CpB1xPjB,C;;QoB4xPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAvqII,8D;MpBvtIJ,iB;MoB+xPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAnrIG,cAAR,iBAAQ,CAAQ,C;QAmrIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBvyPG,MAAO,KoBuyPO,QpBvyPP,EoBuyPiB,CpBvyPjB,C;;QoByyPd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MApsII,8D;MpBnvIJ,iB;MoB2zPA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAhsIG,cAAR,iBAAQ,CAAQ,C;QAgsIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBp0PG,MAAO,KoBo0PO,QpBp0PP,EoBo0PiB,CpBp0PjB,C;;QoBs0Pd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAjtII,8D;MpBhwIJ,iB;MoBw1PA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA7sIG,cAAR,iBAAQ,CAAQ,C;QA6sIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBj2PG,MAAO,KoBi2PO,QpBj2PP,EoBi2PiB,CpBj2PjB,C;;QoBm2Pd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA9tII,8D;MpB7wIJ,iB;MoBq3PA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA1tIG,cAAR,iBAAQ,CAAQ,C;QA0tIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB93PG,MAAO,KoB83PO,QpB93PP,EoB83PiB,CpB93PjB,C;;QoBg4Pd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA3uII,8D;MpB1xIJ,iB;MoBk5PA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAvuIG,cAAR,iBAAQ,CAAQ,C;QAuuIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB35PG,MAAO,KoB25PO,QpB35PP,EoB25PiB,CpB35PjB,C;;QoB65Pd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAxvII,8D;MpBvyIJ,iB;MoB+6PA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OApvIG,cAAR,iBAAQ,CAAQ,C;QAovIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBx7PG,MAAO,KoBw7PO,QpBx7PP,EoBw7PiB,CpBx7PjB,C;;QoB07Pd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MArwII,8D;MpBpzIJ,iB;MoB48PA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAjwIG,cAAR,iBAAQ,CAAQ,C;QAiwIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBr9PG,MAAO,KoBq9PO,QpBr9PP,EoBq9PiB,CpBr9PjB,C;;QoBu9Pd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAlxII,8D;MpBj0IJ,iB;MoBy+PA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9wIG,cAAR,iBAAQ,CAAQ,C;QA8wIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBl/PG,MAAO,KoBk/PO,QpBl/PP,EoBk/PiB,CpBl/PjB,C;;QoBo/Pd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA/xII,8D;MpB91IJ,iB;MoBsgQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA3xIG,cAAR,iBAAQ,CAAQ,C;QA2xIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBhhQG,MAAO,KoBghQO,QpBhhQP,EoBghQiB,CpBhhQjB,C;;QoBkhQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA5yII,8D;MpB32IJ,iB;MoBmiQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAxyIG,cAAR,iBAAQ,CAAQ,C;QAwyIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB7iQG,MAAO,KoB6iQO,QpB7iQP,EoB6iQiB,CpB7iQjB,C;;QoB+iQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAzzII,8D;MpBx3IJ,iB;MoBgkQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OArzIG,cAAR,iBAAQ,CAAQ,C;QAqzIb,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB1kQG,MAAO,KoB0kQO,QpB1kQP,EoB0kQiB,CpB1kQjB,C;;QoB4kQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAt0II,8D;MpBr4IJ,iB;MoB6lQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAl0IG,cAAR,iBAAQ,CAAQ,C;QAk0Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBvmQG,MAAO,KoBumQO,QpBvmQP,EoBumQiB,CpBvmQjB,C;;QoBymQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAn1II,8D;MpBl5IJ,iB;MoB0nQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA/0IG,cAAR,iBAAQ,CAAQ,C;QA+0Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBpoQG,MAAO,KoBooQO,QpBpoQP,EoBooQiB,CpBpoQjB,C;;QoBsoQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAh2II,8D;MpB/5IJ,iB;MoBupQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA52IG,cAAR,iBAAQ,CAAQ,C;QA42Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBjqQG,MAAO,KoBiqQO,QpBjqQP,EoBiqQiB,CpBjqQjB,C;;QoBmqQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA72II,8D;MpB56IJ,iB;MoBorQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAz3IG,cAAR,iBAAQ,CAAQ,C;QAy3Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBhrQG,MAAO,KoBgrQO,QpBhrQP,EoBgrQiB,CpBhrQjB,C;;QoBkrQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA13II,8D;MpBz7IJ,iB;MoBisQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAt4IG,cAAR,iBAAQ,CAAQ,C;QAs4Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBntQG,MAAO,KoBmtQO,QpBntQP,EoBmtQiB,CpBntQjB,C;;QoBqtQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAv4II,8D;MpBt8IJ,iB;MoB8tQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAn5IG,cAAR,iBAAQ,CAAQ,C;QAm5Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBhuQG,MAAO,KoBguQO,QpBhuQP,EoBguQiB,CpBhuQjB,C;;QoBkuQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAp5II,8D;MpBn9IJ,iB;MoB2uQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAh6IG,cAAR,iBAAQ,CAAQ,C;QAg6Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBzvQG,MAAO,KoByvQO,QpBzvQP,EoByvQiB,CpBzvQjB,C;;QoB2vQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAj6II,8D;MpBh+IJ,iB;MoBwwQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA76IG,cAAR,iBAAQ,CAAQ,C;QA66Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBtxQG,MAAO,KoBsxQO,QpBtxQP,EoBsxQiB,CpBtxQjB,C;;QoBwxQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA96II,8D;MpB7+IJ,iB;MoBqyQA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA17IG,cAAR,iBAAQ,CAAQ,C;QA07Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBnzQG,MAAO,KoBmzQO,QpBnzQP,EoBmzQiB,CpBnzQjB,C;;QoBqzQd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MA37II,8D;MpB1/IJ,iB;MoBk0QA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAv8IG,cAAR,iBAAQ,CAAQ,C;QAu8Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpBh1QG,MAAO,KoBg1QO,QpBh1QP,EoBg1QiB,CpBh1QjB,C;;QoBk1Qd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAx8II,8D;MpBvgJJ,iB;MoB+1QA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAp9IG,cAAR,iBAAQ,CAAQ,C;QAo9Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB72QG,MAAO,KoB62QO,QpB72QP,EoB62QiB,CpB72QjB,C;;QoB+2Qd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAr9II,8D;MpBphJJ,iB;MoB43QA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OAj+IG,cAAR,iBAAQ,CAAQ,C;QAi+Ib,aAAU,CAAV,iB;UACI,QAAQ,SAAS,sBAAK,CAAL,CAAT,C;UACR,WpB14QG,MAAO,KoB04QO,QpB14QP,EoB04QiB,CpB14QjB,C;;QoB44Qd,OAAO,Q;O;KAlBX,C;+FAqBA,yB;MAl+II,8D;MpBjiJJ,iB;MoBy5QA,sC;QAciB,Q;QAFb,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,eAAe,SAAS,sBAAK,CAAL,CAAT,C;QACF,OA9+IG,cAAR,iBAAQ,CAAQ,C

S,sBAAK,CAAL,CAAT,C;UACR,IAAI,UAAW,SAAQ,QAAR,EAAkB,CAAlB,CAAX,GAAkC,CAAtC,C;YACI,WAAW,C;;;QAGnB,OAAO,Q;O;KAlBX,C;IAqBA,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA91HG,gBAAR,iBAAQ,C;MA81HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,InC5mP8D,YmC4mP1D,GnC5mP2E,KAAjB,EmC4mPpD,CnC5mPiF,KAA7B,CmC4mP1D,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAr2HG,gBAAR,iBAAQ,C;MAq2HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,InBnnP+D,amBmnP3D,GnBnnP6E,KAAlB,EmBmnPrD,CnBnnPmF,KAA9B,CmBmnP3D,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA52HG,gBAAR,iBAAQ,C;MA42HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IpC1pP4E,0BoC0pPxE,GpC/6O8B,KAAlB,EAAiB,GA3O8B,EoC0pPlE,CpC/6OwB,KAAlB,EAAiB,GA3O8B,CoC0pPxE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAn3HG,gBAAR,iBAAQ,C;MAm3HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAj4HG,gBAAR,iBAAQ,C;MAi4HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA/4HG,gBAAR,iBAAQ,C;MA+4HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA75HG,gBAAR,iBAAQ,C;MA65HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA35HG,gBAAR,iBAAQ,C;MA25HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAz5HG,gBAAR,iBAAQ,C;MAy5HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAv7HG,gBAAR,iBAAQ,C;MAu7HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAr9HG,gBAAR,iBAAQ,C;MAq9HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAn/HG,gBAAR,iBAAQ,C;MAm/HhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAjhIG,gBAAR,iBAAQ,C;MAihIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA3iIG,gBAAR,iBAAQ,C;MA2iIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAvkIG,gBAAR,iBAAQ,C;MAukIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAnmIG,gBAAR,iBAAQ,C;MAmmIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA3nIG,gBAAR,iBAAQ,C;MA2nIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAvoIG,gBAAR,iBAAQ,C;MAuoIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAnpIG,gBAAR,iBAAQ,C;MAmpIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA/qIG,gBAAR,iBAAQ,C;MA+qIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA3rIG,gBAAR,iBAAQ,C;MA2rIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAvsIG,gBAAR,iBAAQ,C;MAusIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAntIG,gBAAR,iBAAQ,C;MAmtIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA3uIG,gBAAR,iBAAQ,C;MA2uIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAvvIG,gBAAR,iBAAQ,C;MAuvIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAnwIG,gBAAR,iBAAQ,C;MAmwIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA3xIG,gBAAR,iBAAQ,C;MA2xIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAvyIG,gBAAR,iBAAQ,C;MAuyIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAnzIG,gBAAR,iBAAQ,C;MAmzIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA3zIG,gBAAR,iBAAQ,C;MA2zIhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAv0IG,gBAAR,iBAAQ,C;MAu0IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAn1IG,gBAAR,iBAAQ,C;MAm1IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA31IG,gBAAR,iBAAQ,C;MA21IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAv2IG,gBAAR,iBAAQ,C;MAu2IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAn3IG,gBAAR,iBAAQ,C;MAm3IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA33IG,gBAAR,iBAAQ,C;MA23IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAv4IG,gBAAR,iBAAQ,C;MAu4IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAn5IG,gBAAR,iBAAQ,C;MAm5IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA35IG,gBAAR,iBAAQ,C;MA25IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAv6IG,gBAAR,iBAAQ,C;MAu6IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAn7IG,gBAAR,iBAAQ,C;MAm7IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA37IG,gBAAR,iBAAQ,C;MA27IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAv8IG,gBAAR,iBAAQ,C;MAu8IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAn9IG,gBAAR,iBAAQ,C;MAm9IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA39IG,gBAAR,iBAAQ,C;MA29IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAv+IG,gBAAR,iBAAQ,C;MAu+IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAn/IG,gBAAR,iBAAQ,C;MAm/IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA3/IG,gBAAR,iBAAQ,C;MA2/IhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAvgJG,gBAAR,iBAAQ,C;MAugJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAnhJG,gBAAR,iBAAQ,C;MAmhJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA3hJG,gBAAR,iBAAQ,C;MA2hJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAviJG,gBAAR,iBAAQ,C;MAuiJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAnjJG,gBAAR,iBAAQ,C;MAmjJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA3jJG,gBAAR,iBAAQ,C;MA2jJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAvkJG,gBAAR,iBAAQ,C;MAukJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAnlJG,gBAAR,iBAAQ,C;MAmlJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA3lJG,gBAAR,iBAAQ,C;MA2lJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAvmJG,gBAAR,iBAAQ,C;MAumJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAAlB,EAAiB,GA3O8B,EkCiqPnE,ClC77OwB,KAAlB,EAAiB,GA3O8B,CkCiqPzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;K;IAGX,iC;MAQiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAnnJG,gBAAR,iBAAQ,C;MAmnJhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IlCjqP6E,0BkCiqPzE,GlC77O8B,KAA

CAAvB,C;;QAElB,OAAO,W;O;KAtBX,C;qFAyBA,yB;MAAA,4F;MAhtII,8D;MAgtIJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAnuID,cAAR,iBAAAQ,C;QAmuIhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;;QAElB,OAAO,W;O;KAtBX,C;qFAyBA,yB;MAAA,4F;MAjuII,8D;MAiuIJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OApvID,cAAR,iBAAAQ,C;QAovIhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAyBA,yB;MAAA,4F;MAlxII,8D;MAkxIJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAryID,cAAR,iBAAAQ,C;QAqyIhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAnyII,8D;MAmyIJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAtzID,cAAR,iBAAAQ,C;QAszIhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MApzII,8D;MAozIJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAv0ID,cAAR,iBAAAQ,C;QAu0IhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAr0II,8D;MAq0IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAx1ID,cAAR,iBAAAQ,C;QAw1IhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAt1II,8D;MAs1IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAz2ID,cAAR,iBAAAQ,C;QAy2IhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAv2II,8D;MAu2IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA14ID,cAAR,iBAAAQ,C;QA04IhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAx4II,8D;MAw4IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA35ID,cAAR,iBAAAQ,C;QA25IhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAz5II,8D;MAy5IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA56ID,cAAR,iBAAAQ,C;QA46IhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA16II,8D;MA06IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA77ID,cAAR,iBAAAQ,C;QA67IhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA37II,8D;MA27IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA98ID,cAAR,iBAAAQ,C;QA88IhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA58II,8D;MA48IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA/9ID,cAAR,iBAAAQ,C;QA+9IhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA79II,8D;MA69IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAh/ID,cAAR,iBAAAQ,C;QAg/IhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA99II,8D;MA89IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAjgJD,cAAR,iBAAAQ,C;QAigJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA/9II,8D;MA+9IJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAlhJD,cAAR,iBAAAQ,C;QAkhJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAhgJI,8D;MAggJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAnjJD,cAAR,iBAAAQ,C;QAmjJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAjhJI,8D;MAihJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OApkJD,cAAR,iBAAAQ,C;QAokJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAlhJI,8D;MAkhJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OArlJD,cAAR,iBAAAQ,C;QAqlJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAnjJI,8D;MAmjJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAtmJD,cAAR,iBAAAQ,C;QAsmJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MApjJI,8D;MAojJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAvnJD,cAAR,iBAAAQ,C;QAunJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MArmJI,8D;MAqmJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAxoJD,cAAR,iBAAAQ,C;QAwoJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAtmJI,8D;MAsmJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAzpJD,cAAR,iBAAAQ,C;QAypJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAvmJI,8D;MAumJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA1qJD,cAAR,iBAAAQ,C;QA0qJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAxpJI,8D;MAwpJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA3rJD,cAAR,iBAAAQ,C;QA2rJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAzpJI,8D;MAypJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA5sJD,cAAR,iBAAAQ,C;QA4sJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA1qJI,8D;MA0qJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA7tJD,cAAR,iBAAAQ,C;QA6tJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA3qJI,8D;MA2qJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA9uJD,cAAR,iBAAAQ,C;QA8uJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA5rJI,8D;MA4rJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA/vJD,cAAR,iBAAAQ,C;QA+vJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA7sJI,8D;MA6sJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAhxJD,cAAR,iBAAAQ,C;QAgxJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA9tJI,8D;MA8tJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAjyJD,cAAR,iBAAAQ,C;QAiyJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA/uJI,8D;MA+uJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAlzJD,cAAR,iBAAAQ,C;QAkzJhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAhwJI,8D;MAgwJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAn0JD,cAAR,iBAAAQ,C;QAm0JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAjxJI,8D;MAixJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAp1JD,cAAR,iBAAAQ,C;QAo1JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAlyJI,8D;MAkyJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAr2JD,cAAR,iBAAAQ,C;QAq2JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAnzJI,8D;MAmzJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAt3JD,cAAR,iBAAAQ,C;QAs3JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MApzJI,8D;MAozJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAv4JD,cAAR,iBAAAQ,C;QAu4JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MArwJI,8D;MAqwJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAx5JD,cAAR,iBAAAQ,C;QAw5JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAtwJI,8D;MAswJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAz6JD,cAAR,iBAAAQ,C;QAy6JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAvxJI,8D;MAuxJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA17JD,cAAR,iBAAAQ,C;QA07JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAxyJI,8D;MAwyJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA38JD,cAAR,iBAAAQ,C;QA28JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAzzJI,8D;MAyzJJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA59JD,cAAR,iBAAAQ,C;QA49JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA10JI,8D;MA00JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA7+JD,cAAR,iBAAAQ,C;QA6+JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA30JI,8D;MA20JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA9/JD,cAAR,iBAAAQ,C;QA8/JhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA51JI,8D;MA41JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA/gKD,cAAR,iBAAAQ,C;QA+gKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA72JI,8D;MA62JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAhiKD,cAAR,iBAAAQ,C;QAgiKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA93JI,8D;MA83JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAjjKD,cAAR,iBAAAQ,C;QAijKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA/3JI,8D;MA+3JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAlkKD,cAAR,iBAAAQ,C;QAkkKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAh5JI,8D;MAg5JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAnlKD,cAAR,iBAAAQ,C;QAmlKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAj6JI,8D;MAi6JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OApmKD,cAAR,iBAAAQ,C;QAomKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAl7JI,8D;MAk7JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OArnKD,cAAR,iBAAAQ,C;QAqnKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAn8JI,8D;MAm8JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAtoKD,cAAR,iBAAAQ,C;QAsoKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAp9JI,8D;MAo9JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAvpKD,cAAR,iBAAAQ,C;QAupKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAr+JI,8D;MAq+JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAxqKD,cAAR,iBAAAQ,C;QAwqKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAt/JI,8D;MAs/JJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAzrKD,cAAR,iBAAAQ,C;QAyrKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAvgKI,8D;MAugKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA1sKD,cAAR,iBAAAQ,C;QA0sKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAxhKI,8D;MAwhKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA3tKD,cAAR,iBAAAQ,C;QA2tKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAziKI,8D;MAyiKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA5uKD,cAAR,iBAAAQ,C;QA4uKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA1jKI,8D;MA0jKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA7vKD,cAAR,iBAAAQ,C;QA6vKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA3kKI,8D;MA2kKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA9wKD,cAAR,iBAAAQ,C;QA8wKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA5lKI,8D;MA4lKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA/xKD,cAAR,iBAAAQ,C;QA+xKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA7mKI,8D;MA6mKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAhzKD,cAAR,iBAAAQ,C;QAgzKhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA9nKI,8D;MA8nKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAj0KD,cAAR,iBAAAQ,C;QAi0KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA/oKI,8D;MA+oKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAl1KD,cAAR,iBAAAQ,C;QAk1KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAhqKI,8D;MAgqKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAn2KD,cAAR,iBAAAQ,C;QAm2KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAjrKI,8D;MAirKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAp3KD,cAAR,iBAAAQ,C;QAo3KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAlsKI,8D;MAksKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAr4KD,cAAR,iBAAAQ,C;QAq4KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAntKI,8D;MAmtKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAt5KD,cAAR,iBAAAQ,C;QAs5KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAptKI,8D;MAotKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAv6KD,cAAR,iBAAAQ,C;QAu6KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MArvKI,8D;MAqvKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAx7KD,cAAR,iBAAAQ,C;QAw7KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAtwKI,8D;MAswKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAz8KD,cAAR,iBAAAQ,C;QAy8KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAvxKI,8D;MAuxKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA19KD,cAAR,iBAAAQ,C;QA09KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAxyKI,8D;MAwyKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA3+KD,cAAR,iBAAAQ,C;QA2+KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MAzzKI,8D;MAyzKJ,uC;QAmBqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA5/KD,cAAR,iBAAAQ,C;QA4/KhB,iBAAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAAuB,sBAAAK,KAAAL,CAAAvB,B,C;mGAAA,yB;MAAA,4F;MA10KI,8D;MA00KJ,uC;QAmBqB,

AO,W;O;KAtBX,C;+FAyBA,yB;MAAA,4F;MA7mJI,8D;MA6mJJ,uC;QAkB0B,UAEU,M;QAJhC,YA7nJgB,cAAR,iBAAQ,C;QA8nJhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,cAAJ,EAAI,sBAAJ,UAAV,EAAwB,WAAxB,C;;QAElB,OAAO,W;O;KAtBX,C;6GAyBA,yB;MAAA,4F;MA9pJI,8D;MA8pJJ,uC;QAkB0B,Q;QAFtB,YA9qJgB,cAAR,iBAAQ,C;QA+qJhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAvBX,C;6GA0BA,yB;MAAA,4F;MAhrJI,8D;MAgrJJ,uC;QAkB0B,Q;QAFtB,YAhsJgB,cAAR,iBAAQ,C;QAisJhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAxBX,C;6GA2BA,yB;MAAA,4F;MAlsJI,8D;MAksJJ,uC;QAkB0B,Q;QAFtB,YAltJgB,cAAR,iBAAQ,C;QAmtJhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA1BX,C;6GA6BA,yB;MAAA,4F;MAptJI,8D;MAotJJ,uC;QAkB0B,Q;QAFtB,YAptJgB,cAAR,iBAAQ,C;QAquJhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA5BX,C;6GA+BA,yB;MAAA,4F;MAtuJI,8D;MAsuJJ,uC;QAkB0B,Q;QAFtB,YAtvJgB,cAAR,iBAAQ,C;QAuvJhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA9BX,C;6GAiCA,yB;MAAA,4F;MAxvJI,8D;MAwvJJ,uC;QAkB0B,Q;QAFtB,YAxwJgB,cAAR,iBAAQ,C;QAywJhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAhCX,C;6GAmCA,yB;MAAA,4F;MA1wJI,8D;MA0wJJ,uC;QAkB0B,Q;QAFtB,YA1xJgB,cAAR,iBAAQ,C;QA2xJhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAlCX,C;6GAqCA,yB;MAAA,4F;MA5xJI,8D;MA4xJJ,uC;QAkB0B,Q;QAFtB,YA5yJgB,cAAR,iBAAQ,C;QA6yJhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KApCX,C;6GAuCA,yB;MAAA,4F;MA9yJI,8D;MA8yJJ,uC;QAkB0B,Q;QAFtB,YA9zJgB,cAAR,iBAAQ,C;QA+zJhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAtCX,C;6GAyCA,yB;MAAA,4F;MAh0JI,8D;MAg0JJ,uC;QAkB0B,Q;QAFtB,YAh1JgB,cAAR,iBAAQ,C;QAi1JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAxCX,C;6GA2CA,yB;MAAA,4F;MAl1JI,8D;MAk1JJ,uC;QAkB0B,Q;QAFtB,YAl2JgB,cAAR,iBAAQ,C;QAm2JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA1CX,C;6GA6CA,yB;MAAA,4F;MAp2JI,8D;MAo2JJ,uC;QAkB0B,Q;QAFtB,YAp3JgB,cAAR,iBAAQ,C;QAq3JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA5CX,C;6GA+CA,yB;MAAA,4F;MAt3JI,8D;MAs3JJ,uC;QAkB0B,Q;QAFtB,YAt4JgB,cAAR,iBAAQ,C;QAu4JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA9CX,C;6GAiDA,yB;MAAA,4F;MAx4JI,8D;MAw4JJ,uC;QAkB0B,Q;QAFtB,YAx5JgB,cAAR,iBAAQ,C;QAy5JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAhDX,C;6GAmDA,yB;MAAA,4F;MA15JI,8D;MA05JJ,uC;QAkB0B,Q;QAFtB,YA16JgB,cAAR,iBAAQ,C;QA26JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAlDX,C;6GAqDA,yB;MAAA,4F;MA56JI,8D;MA46JJ,uC;QAkB0B,Q;QAFtB,YA56JgB,cAAR,iBAAQ,C;QA66JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KApDX,C;6GAuDA,yB;MAAA,4F;MA96JI,8D;MA86JJ,uC;QAkB0B,Q;QAFtB,YA96JgB,cAAR,iBAAQ,C;QA+7JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAtDX,C;6GAyDA,yB;MAAA,4F;MAh8JI,8D;MAg8JJ,uC;QAkB0B,Q;QAFtB,YAh9JgB,cAAR,iBAAQ,C;QAi9JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAxDX,C;6GA2DA,yB;MAAA,4F;MAl9JI,8D;MAk9JJ,uC;QAkB0B,Q;QAFtB,YAl+JgB,cAAR,iBAAQ,C;QAm+JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA1DX,C;6GA6DA,yB;MAAA,4F;MAp+JI,8D;MAo+JJ,uC;QAkB0B,Q;QAFtB,YAp/JgB,cAAR,iBAAQ,C;QAq/JhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA5DX,C;6GA+DA,yB;MAAA,4F;MAt/JI,8D;MAs/JJ,uC;QAkB0B,Q;QAFtB,YAt gKgB,cAAR,iBAAQ,C;QAugKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA9DX,C;6GAiEA,yB;MAAA,4F;MAxgKI,8D;MAwgKJ,uC;QAkB0B,Q;QAFtB,YAxhKgB,cAAR,iBAAQ,C;QAyhKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAhEX,C;6GAmEA,yB;MAAA,4F;MA1hKI,8D;MA0hKJ,uC;QAkB0B,Q;QAFtB,YA1iKgB,cAAR,iBAAQ,C;QA2iKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAlEX,C;6GAqEA,yB;MAAA,4F;MA5iKI,8D;MA4iKJ,uC;QAkB0B,Q;QAFtB,YA5jKgB,cAAR,iBAAQ,C;QA6jKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KApEX,C;6GAuEA,yB;MAAA,4F;MA9jKI,8D;MA8jKJ,uC;QAkB0B,Q;QAFtB,YA9kKgB,cAAR,iBAAQ,C;QA+kKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAtEX,C;6GAyEA,yB;MAAA,4F;MAhlKI,8D;MAglKJ,uC;QAkB0B,Q;QAFtB,YAhmKgB,cAAR,iBAAQ,C;QAimKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAxEX,C;6GA2EA,yB;MAAA,4F;MAlmKI,8D;MAkmKJ,uC;QAkB0B,Q;QAFtB,YAlnKgB,cAAR,iBAAQ,C;QAmnKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA1EX,C;6GA6EA,yB;MAAA,4F;MApnKI,8D;MAonKJ,uC;QAkB0B,Q;QAFtB,YApoKgB,cAAR,iBAAQ,C;QAqoKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA5EX,C;6GA+EA,yB;MAAA,4F;MAtoKI,8D;MAsoKJ,uC;QAkB0B,Q;QAFtB,YAtpKgB,cAAR,iBAAQ,C;QAupKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA9EX,C;6GAiFA,yB;MAAA,4F;MAxpKI,8D;MAwpKJ,uC;QAkB0B,Q;QAFtB,YAxqKgB,cAAR,iBAAQ,C;QAyqKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAhFX,C;6GAmFA,yB;MAAA,4F;MA1qKI,8D;MA0qKJ,uC;QAkB0B,Q;QAFtB,YA1rKgB,cAAR,iBAAQ,C;QA2rKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAlFX,C;6GAqFA,yB;MAAA,4F;MA5rKI,8D;MA4rKJ,uC;QAkB0B,Q;QAFtB,YA5sKgB,cAAR,iBAAQ,C;QA6sKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KApFX,C;6GAuFA,yB;MAAA,4F;MA9sKI,8D;MA8sKJ,uC;QAkB0B,Q;QAFtB,YA9tKgB,cAAR,iBAAQ,C;QA+tKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAtFX,C;6GAyFA,yB;MAAA,4F;MAhuKI,8D;MAguKJ,uC;QAkB0B,Q;QAFtB,YAhvKgB,cAAR,iBAAQ,C;QAivKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAxFX,C;6GA2FA,yB;MAAA,4F;MAlvKI,8D;MAkvKJ,uC;QAkB0B,Q;QAFtB,YAlwKgB,cAAR,iBAAQ,C;QAmwKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA1FX,C;6GA6FA,yB;MAAA,4F;MApwKI,8D;MAowKJ,uC;QAkB0B,Q;QAFtB,YApxKgB,cAAR,iBAAQ,C;QAqxKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA5FX,C;6GA+FA,yB;MAAA,4F;MAtxKI,8D;MAsxKJ,uC;QAkB0B,Q;QAFtB,YAtyKgB,cAAR,iBAAQ,C;QAuyKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA9FX,C;6GAiGA,yB;MAAA,4F;MAxyKI,8D;MAwyKJ,uC;QAkB0B,Q;QAFtB,YAxzKgB,cAAR,iBAAQ,C;QAyzKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAhGX,C;6GAmGA,yB;MAAA,4F;MA1zKI,8D;MA0zKJ,uC;QAkB0B,Q;QAFtB,YA1zKgB,cAAR,iBAAQ,C;QA2zKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAlGX,C;6GAqGA,yB;MAAA,4F;MA5zKI,8D;MA4zKJ,uC;QAkB0B,Q;QAFtB,YA5zKgB,cAAR,iBAAQ,C;QA6zKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KApGX,C;6GAuGA,yB;MAAA,4F;MA9zKI,8D;MA8zKJ,uC;QAkB0B,Q;QAFtB,YA9zKgB,cAAR,iBAAQ,C;QA+zKhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAtGX,C;6GAyGA,yB;MAAA,4F;MAh0KI,8D;MAg0KJ,uC;QAkB0B,Q;QAFtB,YAh0KgB,cAAR,iBAAQ,C;QAi0KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAxGX,C;6GA2GA,yB;MAAA,4F;MAl0KI,8D;MAk0KJ,uC;QAkB0B,Q;QAFtB,YAl0KgB,cAAR,iBAAQ,C;QAm0KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA1GX,C;6GA6GA,yB;MAAA,4F;MAp0KI,8D;MAo0KJ,uC;QAkB0B,Q;QAFtB,YAp0KgB,cAAR,iBAAQ,C;QAq0KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA5GX,C;6GA+GA,yB;MAAA,4F;MAt0KI,8D;MAs0KJ,uC;QAkB0B,Q;QAFtB,YAt0KgB,cAAR,iBAAQ,C;QAu0KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA9GX,C;6GAiHA,yB;MAAA,4F;MAx0KI,8D;MAw0KJ,uC;QAkB0B,Q;QAFtB,YAx0KgB,cAAR,iBAAQ,C;QAy0KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAhHX,C;6GAmHA,yB;MAAA,4F;MA10KI,8D;MA00KJ,uC;QAkB0B,Q;QAFtB,YA10KgB,cAAR,iBAAQ,C;QA20KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAlHX,C;6GAqHA,yB;MAAA,4F;MA50KI,8D;MA40KJ,uC;QAkB0B,Q;QAFtB,YA50KgB,cAAR,iBAAQ,C;QA60KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KApHX,C;6GAuHA,yB;MAAA,4F;MA90KI,8D;MA80KJ,uC;QAkB0B,Q;QAFtB,YA90KgB,cAAR,iBAAQ,C;QA+0KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAtHX,C;6GAyHA,yB;MAAA,4F;MAh1KI,8D;MAg1KJ,uC;QAkB0B,Q;QAFtB,YAh1KgB,cAAR,iBAAQ,C;QAi1KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAxHX,C;6GA2HA,yB;MAAA,4F;MAl1KI,8D;MAk1KJ,uC;QAkB0B,Q;QAFtB,YAl1KgB,cAAR,iBAAQ,C;QAm1KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA1HX,C;6GA6HA,yB;MAAA,4F;MAp1KI,8D;MAo1KJ,uC;QAkB0B,Q;QAFtB,YAp1KgB,cAAR,iBAAQ,C;QAq1KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA5HX,C;6GA+HA,yB;MAAA,4F;MAt1KI,8D;MAs1KJ,uC;QAkB0B,Q;QAFtB,YAt1KgB,cAAR,iBAAQ,C;QAu1KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KA9HX,C;6GAiIA,yB;MAAA,4F;MAx1KI,8D;MAw1KJ,uC;QAkB0B,Q;QAFtB,YAx1KgB,cAAR,iBAAQ,C;QAy1KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAhIX,C;6GAmIA,yB;MAAA,4F;MA11KI,8D;MA01KJ,uC;QAkB0B,Q;QAFtB,YA11KgB,cAAR,iBAAQ,C;QA21KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KAlIX,C;6GAqIA,yB;MAAA,4F;MA51KI,8D;MA41KJ,uC;QAkB0B,Q;QAFtB,YA51KgB,cAAR,iBAAQ,C;QA61KhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAAkB,uBAAI,YAAJ,EAAI,oBAAJ,Q;QAClB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,WAAV,EAAwB,sBAAI,KAAJ,CAAjB,EAAwB,EAAxB,C;UACd,qB;;QAEJ,OAAO,W;O;KApIX,C;6GAuIA,yB;MAAA,4F;MA91KI,8D;MA81KJ,uC;QAkB0B,Q;QAFtB,YA91KgB,cAAR,iBAAQ,C;

P,C;QACc,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,arBvwRO,W;QqBwwRP,kBAAkB,O;QACF,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;+FAyBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAkBoB,Q;QAAhB,IAAI,mBAAJ,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,arBhyRO,W;QqBiyRP,kBAAkB,O;QACF,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;6GA0BA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAAA,0D;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAvBX,C;6GA2BA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAkBoB,Q;QAAhB,IAAI,mBAAJ,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,arBp1RO,W;QqBq1RP,kBAAkB,O;QACF,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;6GA0BA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAAA,0D;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAvBX,C;6GA2BA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAkBoB,Q;QAAhB,IAAI,mBAAJ,C;UAAe,OAAO,OAAO,OAAP,C;QACc,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;QAA+B,8B;QAA5C,arB92RO,W;QqB+2RP,kBAAkB,O;QACF,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAtBX,C;6GA0BA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAAA,0D;UACI,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;UACd,MAAO,WAAI,WAAJ,C;;QAEX,OAAO,M;O;KAvBX,C;6GA2

W;QACtB,sBAAkB,sBAAK,CAAL,CAAlB,C;QACqC,kBAAxB,eAAkB,cAAlB,C;QAAgC,sBAAI,aAAJ,C;QAA
7C,arBrlSO,W;QqBslSe,qB;QAAtB,iBAAc,CAAd,wB;UACI,gBAAc,UAAU,KAAV,EAAiB,aAAjB,EAA8B,sBA
AK,KAAL,CAA9B,C;UACd,MAAO,WAAI,aAAJ,C;;QAEX,OAAO,M;O;KAvBX,C;iFA0BA,yB;MAxZA,gD;M
AAA,gE;MAwZA,gD;QAgBW,sB;;UAtZS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAyZH,OAzZG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAwZzB,OAxZyB,C;UAA5C,arBrtRO,W;UqBstR
P,kBAuZmB,O;UAtZH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAqZwB,SArZV,CAAU,WAAV,EAAu
B,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAkZP,yB;O;KAhBJ,C;iFAmBA,yB;MAlZA,g
D;MAAA,gE;MAkZA,gD;QAgBW,sB;;UAhZS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAmZH,OAnZG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAkZzB,OAlZyB,C;UAA5C,arB9uRO,W;Uq
B+uRP,kBAiZmB,O;UAhZH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA+YwB,SA/YV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA4YP,yB;O;KAhBJ,C;iFAmBA,yB;M
A5YA,gD;MAAA,gE;MA4YA,gD;QAgBW,sB;;UA1YS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA6YH,OA7YG,
C;YAAP,uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA4YzB,OA5YyB,C;UAA5C,arBvw
RO,W;UqBwwRP,kBA2YmB,O;UA1YH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAyXwB,SAzXV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAsXP,yB;O;KAhBJ,C;iF
AmBA,yB;MAtXA,gD;MAAA,gE;MAsXA,gD;QAgBW,sB;;UApXS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAuXH,OAAG,
C;YAAP,uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAsXzB,OAtXyB,C;UA5YA,arBhy
RO,W;UqBiyRP,kBAqXmB,O;UApXH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAmWwB,SAnWV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAgWP,yB;O;KAhBJ,C;+FAmBA,yB;MAhWA,gD;MAAA,gE;MAgWA,gD;QAgBW,sB;;UAlVS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAqWH,OArWG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAoWzB,OApWyB,C;UAA5C,arBp1RO,W;UqBq1RP,kBAmWmB,O;UAlWH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAiVwB,SAjVV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA8UP,yB;O;KAhBJ,C;+FAmBA,yB;MA9UA,gD;MAAA,gE;MA8UA,gD;QAgBW,sB;;UAhUS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAmVH,OAnVG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAkVzB,OAlVyB,C;UAA5C,arBx4RO,W;UqBy4RP,kBAiVmB,O;UAhVH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA+TwB,SA/TV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA4TP,yB;O;KAhBJ,C;+FAmBA,yB;MA5TA,gD;MAAA,gE;MA4TA,gD;QAgBW,sB;;UA9SS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAiUH,OAjUG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAgUzB,OAhUyB,C;UAA5C,arB14RO,W;UqB24RP,kBA+TmB,O;UA9TH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA6SwB,SA7SV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA0SP,yB;O;KAhBJ,C;+FAmBA,yB;MA1SA,gD;MAAA,gE;MA0SA,gD;QAgBW,sB;;UA5RS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA+TH,OA/TG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA8TzB,OA9TyB,C;UAA5C,arB54RO,W;UqB64RP,kBA6TmB,O;UA5TH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA2SwB,SA3SV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAwSP,yB;O;KAhBJ,C;+FAmBA,yB;MAxSA,gD;MAAA,gE;MAwSA,gD;QAgBW,sB;;UA1RS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA6TH,OA7TG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA4TzB,OA5TyB,C;UAA5C,arB94RO,W;UqB+4RP,kBA2TmB,O;UA1TH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAyRwB,SAzRV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAsRP,yB;O;KAhBJ,C;+FAmBA,yB;MAtRA,gD;MAAA,gE;MAsRA,gD;QAgBW,sB;;UAxQS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA2SH,OA3SG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA0SzB,OA1SyB,C;UAA5C,arBh5RO,W;UqBi5RP,kBAySmB,O;UAxSH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAuRwB,SAvRV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAoRP,yB;O;KAhBJ,C;+FAmBA,yB;MApRA,gD;MAAA,gE;MAoRA,gD;QAgBW,sB;;UAtQS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAyRH,OAzRG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAwRzB,OAxRyB,C;UAA5C,arBl5RO,W;UqBm5RP,kBAuRmB,O;UAtRH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAqQwB,SArQV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAkQP,yB;O;KAhBJ,C;+FAmBA,yB;MAlQA,gD;MAAA,gE;MAkQA,gD;QAgBW,sB;;UApPS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAuQH,OAvQG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAsQzB,OAtQyB,C;UAA5C,arBp5RO,W;UqBq5RP,kBAqQmB,O;UApQH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAmPwB,SAnPV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAgPP,yB;O;KAhBJ,C;+FAmBA,yB;MAhPA,gD;MAAA,gE;MAgPA,gD;QAgBW,sB;;UAlOS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAqPH,OArPG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAoPzB,OApPyB,C;UAA5C,arBt5RO,W;UqBu5RP,kBAmPmB,O;UAlPH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAiOwB,SAjOV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA8NP,yB;O;KAhBJ,C;+FAmBA,yB;MA9NA,gD;MAAA,gE;MA8NA,gD;QAgBW,sB;;UAhNS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAmOH,OAnOG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAkOzB,OAlOyB,C;UAA5C,arBx5RO,W;UqBy5RP,kBAiOmB,O;UAhOH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA+MwB,SA/MV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA4MP,yB;O;KAhBJ,C;+FAmBA,yB;MA5MA,gD;MAAA,gE;MA4MA,gD;QAgBW,sB;;UA9LS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAiNH,OAjNG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAgNzB,OAhNyB,C;UAA5C,arB15RO,W;UqB25RP,kBA+MmB,O;UA9MH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA6LwB,SA7LV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA0LP,yB;O;KAhBJ,C;+FAmBA,yB;MA1LA,gD;MAAA,gE;MA0LA,gD;QAgBW,sB;;UA5KS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA+MH,OA/MG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA8MzB,OA9MyB,C;UAA5C,arB54RO,W;UqB65RP,kBA6MmB,O;UA5MH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA2LwB,SA3LV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAwLP,yB;O;KAhBJ,C;+FAmBA,yB;MAxLA,gD;MAAA,gE;MAwLA,gD;QAgBW,sB;;UA1KS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA6MH,OA7MG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA4MzB,OA5MyB,C;UAA5C,arB94RO,W;UqB+4RP,kBA2MmB,O;UA1MH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAyLwB,SAzLV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAsLP,yB;O;KAhBJ,C;+FAmBA,yB;MAtLA,gD;MAAA,gE;MAsLA,gD;QAgBW,sB;;UAxKS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA2MH,OA3MG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA0MzB,OA1MyB,C;UAA5C,arBh5RO,W;UqBi5RP,kBAyMmB,O;UAxMH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAuLwB,SAvLV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAoLP,yB;O;KAhBJ,C;+FAmBA,yB;MApLA,gD;MAAA,gE;MAoLA,gD;QAgBW,sB;;UAtKS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAyMH,OAzMG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAwMzB,OAxMyB,C;UAA5C,arBl5RO,W;UqBm5RP,kBAuMmB,O;UAtMH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAqKwB,SArKV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAkKP,yB;O;KAhBJ,C;+FAmBA,yB;MAlKA,gD;MAAA,gE;MAkKA,gD;QAgBW,sB;;UApJS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAuKH,OAvKG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAsKzB,OAtKyB,C;UAA5C,arBp5RO,W;UqBq5RP,kBAqKmB,O;UApKH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAmJwB,SAnJV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAgJP,yB;O;KAhBJ,C;+FAmBA,yB;MAhJA,gD;MAAA,gE;MAgJA,gD;QAgBW,sB;;UAlIS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAqJH,OArJG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAoJzB,OApJyB,C;UAA5C,arBt5RO,W;UqBu5RP,kBAmJmB,O;UAlJH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAiIwB,SAjIV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA8HP,yB;O;KAhBJ,C;+FAmBA,yB;MA9HA,gD;MAAA,gE;MA8HA,gD;QAgBW,sB;;UAhHS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAmIH,OAnIG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAkIzB,OAlIyB,C;UAA5C,arBx5RO,W;UqBy5RP,kBAiImB,O;UAhIH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA+GwB,SA/GV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA4GP,yB;O;KAhBJ,C;+FAmBA,yB;MA5GA,gD;MAAA,gE;MA4GA,gD;QAgBW,sB;;UA9FS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAiHH,OAjHG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAgHzB,OAhHyB,C;UAA5C,arB15RO,W;UqB25RP,kBA+GmB,O;UA9GH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA6FwB,SA7FV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA0FP,yB;O;KAhBJ,C;+FAmBA,yB;MA1FA,gD;MAAA,gE;MA0FA,gD;QAgBW,sB;;UA5ES,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA+GH,OA/GG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA8GzB,OA9GyB,C;UAA5C,arB54RO,W;UqB65RP,kBA6GmB,O;UA5GH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA2FwB,SA3FV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAwFP,yB;O;KAhBJ,C;+FAmBA,yB;MAxFA,gD;MAAA,gE;MAwFA,gD;QAgBW,sB;;UA1ES,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA6GH,OA7GG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA4GzB,OA5GyB,C;UAA5C,arB94RO,W;UqB+4RP,kBA2GmB,O;UA1GH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAyFwB,SAzFV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAsFP,yB;O;KAhBJ,C;+FAmBA,yB;MAtFA,gD;MAAA,gE;MAsFA,gD;QAgBW,sB;;UAxES,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA2GH,OA3GG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA0GzB,OA1GyB,C;UAA5C,arBh5RO,W;UqBi5RP,kBAyGmB,O;UAxGH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAuFwB,SAvFV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAoFP,yB;O;KAhBJ,C;+FAmBA,yB;MApFA,gD;MAAA,gE;MAoFA,gD;QAgBW,sB;;UAtES,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAyGH,OAzGG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAwGzB,OAxGyB,C;UAA5C,arBl5RO,W;UqBm5RP,kBAuGmB,O;UAtGH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAqFwB,SArFV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAkFP,yB;O;KAhBJ,C;+FAmBA,yB;MAlFA,gD;MAAA,gE;MAkFA,gD;QAgBW,sB;;UApES,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAuGH,OAvGG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAsGzB,OAtGyB,C;UAA5C,arBp5RO,W;UqBq5RP,kBAqGmB,O;UApGH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAmFwB,SAnFV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAgFP,yB;O;KAhBJ,C;+FAmBA,yB;MAhFA,gD;MAAA,gE;MAgFA,gD;QAgBW,sB;;UAlES,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAqGH,OArGG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAoGzB,OApGyB,C;UAA5C,arBt5RO,W;UqBu5RP,kBAmGmB,O;UAlGH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAiFwB,SAjFV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA8EP,yB;O;KAhBJ,C;+FAmBA,yB;MA9EA,gD;MAAA,gE;MA8EA,gD;QAgBW,sB;;UAhES,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAmGH,OAnGG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAkGzB,OAlGyB,C;UAA5C,arBx5RO,W;UqBy5RP,kBAiGmB,O;UAhGH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA+EwB,SA/EV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA4EP,yB;O;KAhBJ,C;+FAmBA,yB;MA5EA,gD;MAAA,gE;MA4EA,gD;QAgBW,sB;;UA9DS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAiFH,OAjFG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAgFzB,OAhFyB,C;UAA5C,arB15RO,W;UqB25RP,kBA+EmB,O;UA9EH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA6DwB,SA7DV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QA0DP,yB;O;KAhBJ,C;+FAmBA,yB;MA1DA,gD;MAAA,gE;MA0DA,gD;QAgBW,sB;;UA5CS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA+EH,OA/EG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA8EzB,OA9EyB,C;UAA5C,arB54RO,W;UqB65RP,kBA6EmB,O;UA5EH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cA2CwB,SA3CV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAwCP,yB;O;KAhBJ,C;+FAmBA,yB;MAxCA,gD;MAAA,gE;MAwCA,gD;QAgBW,sB;;UA1BS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA6EH,OA7EG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA4EzB,OA5EyB,C;UAA5C,arB94RO,W;UqB+4RP,kBA2EmB,O;UA1EH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAyBwB,SAzBV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAsBP,yB;O;KAhBJ,C;+FAmBA,yB;MAtBA,gD;MAAA,gE;MAsBA,gD;QAgBW,sB;;UAxBS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA2EH,OA3EG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA0EzB,OA1EyB,C;UAA5C,arBh5RO,W;UqBi5RP,kBAyEmB,O;UAxEH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAuBwB,SAvBV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAoBP,yB;O;KAhBJ,C;+FAmBA,yB;MApBA,gD;MAAA,gE;MAoBA,gD;QAgBW,sB;;UAtBS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAyEH,OAzEG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAwEzB,OAxEyB,C;UAA5C,arBl5RO,W;UqBm5RP,kBAuEmB,O;UAtEH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAqBwB,SArBV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAkBP,yB;O;KAhBJ,C;+FAmBA,yB;MAlBA,gD;MAAA,gE;MAkBA,gD;QAgBW,sB;;UApBS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAuEH,OAvEG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAsEzB,OAtEyB,C;UAA5C,arBp5RO,W;UqBq5RP,kBAqEmB,O;UApEH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAmBwB,SAnBV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAgBP,yB;O;KAhBJ,C;+FAmBA,yB;MAhBA,gD;MAAA,gE;MAgBA,gD;QAgBW,sB;;UAlBS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAqEH,OArEG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAoEzB,OApEyB,C;UAA5C,arBt5RO,W;UqBu5RP,kBAmEmB,O;UAlEH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAiBwB,SAjBV,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAcP,yB;O;KAhBJ,C;+FAmBA,yB;MAdA,gD;MAAA,gE;MAcA,gD;QAgBW,sB;;UAhBS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAmEH,OAnEG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAkEzB,OAlEyB,C;UAA5C,arBx5RO,W;UqBy5RP,kBAiEmB,O;UAhEH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAeuB,SAfT,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAYP,yB;O;KAhBJ,C;+FAmBA,yB;MAZA,gD;MAAA,gE;MAYA,gD;QAgBW,sB;;UAdS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAiEH,OAjEG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAgEzB,OAhEyB,C;UAA5C,arB15RO,W;UqB25RP,kBA+DmB,O;UA9DH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAaqB,SAbP,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAUP,yB;O;KAhBJ,C;+FAmBA,yB;MAVA,gD;MAAA,gE;MAUA,gD;QAgBW,sB;;UAZS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA+DH,OA/DG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA8DzB,OA9DyB,C;UAA5C,arB54RO,W;UqB65RP,kBA6DmB,O;UA5DH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAWqB,SAXP,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAQP,yB;O;KAhBJ,C;+FAmBA,yB;MARA,gD;MAAA,gE;MAQA,gD;QAgBW,sB;;UAVS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA6DH,OA7DG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA4DzB,OA5DyB,C;UAA5C,arB94RO,W;UqB+4RP,kBA2DmB,O;UA1DH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cASqB,SATP,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAMP,yB;O;KAhBJ,C;+FAmBA,yB;MANA,gD;MAAA,gE;MAMA,gD;QAgBW,sB;;UARS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OA2DH,OA3DG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBA0DzB,OA1DyB,C;UAA5C,arBh5RO,W;UqBi5RP,kBAyDmB,O;UAxDH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAOqB,SAPP,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAIP,yB;O;KAhBJ,C;+FAmBA,yB;MAJA,gD;MAAA,gE;MAIA,gD;QAgBW,sB;;UANS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAyDH,OAzDG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAwDzB,OAxDyB,C;UAA5C,arBx5RO,W;UqBy5RP,kBAuDmB,O;UAtDH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAKqB,SALP,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAEP,yB;O;KAhBJ,C;+FAmBA,yB;MAFA,gD;MAAA,gE;MAEA,gD;QAgBW,sB;;UAJS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAuDH,OAvDG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAsDzB,OAtDyB,C;UAA5C,arBp5RO,W;UqBq5RP,kBAqDmB,O;UApDH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAGqB,SAHP,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAAP,yB;O;KAhBJ,C;+FAmBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAgBW,sB;;UAFS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAqDH,OArDG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAoDzB,OApDyB,C;UAA5C,arBt5RO,W;UqBu5RP,kBAmDmB,O;UAlDH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAAqB,SAAP,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAAP,yB;O;KAhBJ,C;+FAmBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAgBW,sB;;UAAS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAmDH,OAnDG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAkDzB,OAlDyB,C;UAA5C,arBx5RO,W;UqBy5RP,kBAiDmB,O;UAhDH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ,cAAqB,SAAP,CAAU,WAAV
,EAAuB,OAAvB,C;YACd,MAAO,WAAI,WAAJ,C;;UAEX,qBAAO,M;;;QAAP,yB;O;KAhBJ,C;+FAmBA,yB;MAAA,gD;MAAA,gE;MAAA,gD;QAgBW,sB;;UAAS,Q;UAHhB,IAAI,mBAAJ,C;YAAe,qBAAO,OAiDH,OAjDG,C;YAAP,
uB;;UACqB,kBAAvB,eAAa,iBAAO,CAAP,IAAb,C;UAA+B,sBAgDzB,OAhDyB,C;UAA5C,arB15RO,W;UqB25RP,kBA+CmB,O;UA9CH,2B;UAAhB,OAAgB,cAAhB,C;YAAgB,yB;YACZ

QAAgB,yB;QACZ,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;K;+FAGX,+B;MAUoB,Q;MADhB,UAAkB,G;MACF,2B;MAAhB,OAAgB,cAAhB,C;

P,sBAAK,CAAL,CA5DO,EAAnB,KA4DqB,CAAM,CAAN,CA5DF,CA4DrB,C;;MA5DT,OA8DO,I;K;IA3DX,kC;MAyEI,WpBzpTO,MAAO,KoBypTG,cpBzpTH,EoBylTH,KAgEkB,OpBzpTf,C;MoB0pTd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAnEqB,GAmEP,sBAAK,CAAL,CAnEO,EAAnB,KAmEqB,CAAM,CAAN,CAnEF,CAmEQ,C;;MAnET,OAqEO,I;K;IAlEX,kC;MAgFI,WpB5qTO,MAAO,KoB4qTG,cpB5qTH,EoBqmTH,KAuEkB,OpB5qTf,C;MoB6qTd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA1EqB,GA0EP,sBAAK,CAAL,CA1EO,EAAnB,KA0EqB,CAAM,CAAN,CA1EF,CA0EQ,C;;MA1ET,OA4EO,I;K;+EAzEX,yB;MA0FI,MAAA,gE;MpB9mTA,iB;MoB8mTA,8C;QAWI,WpBnnTO,MAAO,KoBmnTG,cpBnnTH,EoBmmTH,KAwEkB,OpBnnTf,C;QoBonTd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0GI,MAAA,gE;MpBjpTA,iB;MoBipTA,8C;QAWI,WpBtpTO,MAAO,KoBspTG,cpBtpTH,EoBsoTH,KA2FkB,OpBtpTf,C;QoBupTd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0HI,MAAA,gE;MpBpqTA,iB;MoBoqTA,8C;QAWI,WpBzqTO,MAAO,KoByqTG,cpBzqTH,EoBypTH,KA8GkB,OpBzqTf,C;QoB0qTd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0II,MAAA,gE;MpBvrTA,iB;MoBurTA,8C;QAWI,WpB5rTO,MAAO,KoB4rTG,cpB5rTH,EoB4qTH,KAiIkB,OpB5rTf,C;QoB6rTd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0JI,MAAA,gE;MpB1sTA,iB;MoB0sTA,8C;QAWI,WpB/sTO,MAAO,KoB+sTG,cpB/sTH,EoB+rTH,KAoJkB,OpB/sTf,C;QoBgtTd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0KI,MAAA,gE;MpB7tTA,iB;MoB6tTA,8C;QAWI,WpBluTO,MAAO,KoBkuTG,cpBluTH,EoBksTH,KAuKkB,OpBluTf,C;QoBmuTd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0LI,MAAA,gE;MpBhvTA,iB;MoBgvTA,8C;QAWI,WpBrvTO,MAAO,KoBqvTG,cpBrvTH,EoBqtTH,KA0LkB,OpBrvTf,C;QoBsvTd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0MI,MAAA,gE;MpBnwTA,iB;MoBmwTA,8C;QAWI,WpBxwTO,MAAO,KoBwwTG,cpBxwTH,EoBwuTH,KA6MkB,OpBxwTf,C;QoBywTd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0NI,MAAA,gE;MpBtxTA,iB;MoBsxTA,8C;QAWI,WpB3xTO,MAAO,KoB2xTG,cpB3xTH,EoB2vTH,KAgNkB,OpB3xTf,C;QoB4xTd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0OI,MAAA,gE;MpBzyTA,iB;MoByyTA,8C;QAWI,WpB9yTO,MAAO,KoB8yTG,cpB9yTH,EoB8wTH,KAmOkB,OpB9yTf,C;QoB+yTd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0PI,MAAA,gE;MpB5zTA,iB;MoB4zTA,8C;QAWI,WpBj0TO,MAAO,KoBi0TG,cpBj0TH,EoBixTH,KAsPkB,OpBj0Tf,C;QoBk0Td,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0QI,MAAA,gE;MpB/0TA,iB;MoB+0TA,8C;QAWI,WpBp1TO,MAAO,KoBo1TG,cpBp1TH,EoBoyTH,KAyQkB,OpBp1Tf,C;QoBq1Td,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0RI,MAAA,gE;MpBl2TA,iB;MoBk2TA,8C;QAWI,WpBv2TO,MAAO,KoBu2TG,cpBv2TH,EoBuzTH,KA4RkB,OpBv2Tf,C;QoBw2Td,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0SI,MAAA,gE;MpBr3TA,iB;MoBq3TA,8C;QAWI,WpB13TO,MAAO,KoB03TG,cpB13TH,EoB00TH,KA+SkB,OpB13Tf,C;QoB23Td,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0TI,MAAA,gE;MpBx4TA,iB;MoBw4TA,8C;QAWI,WpB74TO,MAAO,KoB64TG,cpB74TH,EoB61TH,KAkTkB,OpB74Tf,C;QoB84Td,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0UI,MAAA,gE;MpB35TA,iB;MoB25TA,8C;QAWI,WpBh6TO,MAAO,KoBg6TG,cpBh6TH,EoBg3TH,KAqUkB,OpBh6Tf,C;QoBi6Td,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0VI,MAAA,gE;MpB96TA,iB;MoB86TA,8C;QAWI,WpBn7TO,MAAO,KoBm7TG,cpBn7TH,EoBm4TH,KAwVkB,OpBn7Tf,C;QoBo7Td,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0WI,MAAA,gE;MpBj8TA,iB;MoBi8TA,8C;QAWI,WpBt8TO,MAAO,KoBs8TG,cpBt8TH,EoBs5TH,KA2WkB,OpBt8Tf,C;QoBu8Td,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0XI,MAAA,gE;MpBp9TA,iB;MoBo9TA,8C;QAWI,WpBz9TO,MAAO,KoBy9TG,cpBz9TH,EoBy6TH,KA8XkB,OpBz9Tf,C;QoB09Td,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0YI,MAAA,gE;MpBv+TA,iB;MoBu+TA,8C;QAWI,WpB5+TO,MAAO,KoB4+TG,cpB5+TH,EoB47TH,KAiYkB,OpB5+Tf,C;QoB6+Td,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0ZI,MAAA,gE;MpB1/TA,iB;MoB0/TA,8C;QAWI,WpB//TO,MAAO,KoB+/TG,cpB//TH,EoB+8TH,KAoZkB,OpB//Tf,C;QoBggUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0aI,MAAA,gE;MpB7gUA,iB;MoB6gUA,8C;QAWI,WpBlhUO,MAAO,KoBkhUG,cpBlhUH,EoBk+TH,KAuakB,OpBlhUf,C;QoBmhUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0bI,MAAA,gE;MpBhiUA,iB;MoBgiUA,8C;QAWI,WpBriUO,MAAO,KoBqiUG,cpBriUH,EoBq/TH,KA0bkB,OpBriUf,C;QoBsiUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0cI,MAAA,gE;MpBnjUA,iB;MoBmjUA,8C;QAWI,WpBxjUO,MAAO,KoBwjUG,cpBxjUH,EoBwgUH,KA6bkB,OpBxjUf,C;QoByjUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0dI,MAAA,gE;MpBtkUA,iB;MoBskUA,8C;QAWI,WpB3kUO,MAAO,KoB2kUG,cpB3kUH,EoB2hUH,KAgckB,OpB3kUf,C;QoB4kUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0eI,MAAA,gE;MpBzlUA,iB;MoBylUA,8C;QAWI,WpB9lUO,MAAO,KoB8lUG,cpB9lUH,EoB8iUH,KAmekB,OpB9lUf,C;QoB+lUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0fI,MAAA,gE;MpB5mUA,iB;MoB4mUA,8C;QAWI,WpBjnUO,MAAO,KoBinUG,cpBjnUH,EoBikUH,KAsfkB,OpBjnUf,C;QoBknUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0gBI,MAAA,gE;MpB/nUA,iB;MoB+nUA,8C;QAWI,WpBpoUO,MAAO,KoBooUG,cpBpoUH,EoBolUH,KAygBkB,OpBpoUf,C;QoBqoUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0hBI,MAAA,gE;MpBlpUA,iB;MoBkpUA,8C;QAWI,WpBvpUO,MAAO,KoBupUG,cpBvpUH,EoBumUH,KA4hBkB,OpBvpUf,C;QoBwpUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0iBI,MAAA,gE;MpBrqUA,iB;MoBqqUA,8C;QAWI,WpB1qUO,MAAO,KoB0qUG,cpB1qUH,EoB0nUH,KA+iBkB,OpB1qUf,C;QoB2qUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0jBI,MAAA,gE;MpBxrUA,iB;MoBwrUA,8C;QAWI,WpB7rUO,MAAO,KoB6rUG,cpB7rUH,EoB6oUH,KAkkBkB,OpB7rUf,C;QoB8rUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0kBI,MAAA,gE;MpB3sUA,iB;MoB2sUA,8C;QAWI,WpBhtUO,MAAO,KoBgtUG,cpBhtUH,EoBgqUH,KAqlBkB,OpBhtUf,C;QoBitUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0lBI,MAAA,gE;MpB9tUA,iB;MoB8tUA,8C;QAWI,WpBnuUO,MAAO,KoBmuUG,cpBnuUH,EoBmrUH,KAwmBkB,OpBnuUf,C;QoBouUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0nBI,MAAA,gE;MpBjvUA,iB;MoBivUA,8C;QAWI,WpBtvUO,MAAO,KoBsvUG,cpBtvUH,EoBssUH,KA2nBkB,OpBtvUf,C;QoBuvUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0oBI,MAAA,gE;MpBpwUA,iB;MoBowUA,8C;QAWI,WpBzwUO,MAAO,KoBywUG,cpBzwUH,EoBytUH,KA8oBkB,OpBzwUf,C;QoB0wUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0pBI,MAAA,gE;MpBvxUA,iB;MoBuxUA,8C;QAWI,WpB5xUO,MAAO,KoB4xUG,cpB5xUH,EoB4uUH,KAiqBkB,OpB5xUf,C;QoB6xUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0qBI,MAAA,gE;MpB1yUA,iB;MoB0yUA,8C;QAWI,WpB/yUO,MAAO,KoB+yUG,cpB/yUH,EoB+vUH,KAoqBkB,OpB/yUf,C;QoBgzUd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0rBI,MAAA,gE;MpB70UA,iB;MoB60UA,8C;QAWI,WpBl1UO,MAAO,KoBk1UG,cpBl1UH,EoBkxUH,KAurBkB,OpBl1Uf,C;QoBm1Ud,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0sBI,MAAA,gE;MpBh3UA,iB;MoBg3UA,8C;QAWI,WpBr3UO,MAAO,KoBq3UG,cpBr3UH,EoBqzUH,KA0sBkB,OpBr3Uf,C;QoBs3Ud,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0tBI,MAAA,gE;MpBn4UA,iB;MoBm4UA,8C;QAWI,WpBx4UO,MAAO,KoBw4UG,cpBx4UH,EoBw1UH,KA6tBkB,OpBx4Uf,C;QoBy4Ud,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0uBI,MAAA,gE;MpBt5UA,iB;MoBs5UA,8C;QAWI,WpB35UO,MAAO,KoB25UG,cpB35UH,EoB23UH,KAgvBkB,OpB35Uf,C;QoB45Ud,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0vBI,MAAA,gE;MpBz6UA,iB;MoBy6UA,8C;QAWI,WpB96UO,MAAO,KoB86UG,cpB96UH,EoB84UH,KAmwBkB,OpB96Uf,C;QoB+6Ud,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0wBI,MAAA,gE;MpB57UA,iB;MoB47UA,8C;QAWI,WpBj8UO,MAAO,KoBi8UG,cpBj8UH,EoBi6UH,KAswBkB,OpBj8Uf,C;QoBk8Ud,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0xBI,MAAA,gE;MpB/8UA,iB;MoB+8UA,8C;QAWI,WpBp9UO,MAAO,KoBo9UG,cpBp9UH,EoBo7UH,KAyxBkB,OpBp9Uf,C;QoBq9Ud,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0yBI,MAAA,gE;MpBl+UA,iB;MoBk+UA,8C;QAWI,WpBv+UO,MAAO,KoBu+UG,cpBv+UH,EoBu8UH,KA4yBkB,OpBv+Uf,C;QoBw+Ud,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0zBI,MAAA,gE;MpBr/UA,iB;MoBq/UA,8C;QAWI,WpB1/UO,MAAO,KoB0/UG,cpB1/UH,EoB09UH,KA+zBkB,OpB1/Uf,C;QoB2/Ud,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA00BI,MAAA,gE;MpBxgVA,iB;MoBwgVA,8C;QAWI,WpB7gVO,MAAO,KoB6gVG,cpB7gVH,EoB6+UH,KAk0BkB,OpB7gVf,C;QoB8gVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA01BI,MAAA,gE;MpB3hVA,iB;MoB2hVA,8C;QAWI,WpBhiVO,MAAO,KoBgiVG,cpBhiVH,EoBg/UH,KAq1BkB,OpBhiVf,C;QoBiiVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA02BI,MAAA,gE;MpB9iVA,iB;MoB8iVA,8C;QAWI,WpBnjVO,MAAO,KoBmjVG,cpBnjVH,EoBmgVH,KAw2BkB,OpBnjVf,C;QoBojVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA03BI,MAAA,gE;MpBjkVA,iB;MoBikVA,8C;QAWI,WpBtkVO,MAAO,KoBskVG,cpBtkVH,EoBshVH,KA23BkB,OpBtkVf,C;QoBukVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA04BI,MAAA,gE;MpBpmVA,iB;MoBomVA,8C;QAWI,WpBzmVO,MAAO,KoBymVG,cpBzmVH,EoByiVH,KA84BkB,OpBzmVf,C;QoB0mVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA05BI,MAAA,gE;MpBvnVA,iB;MoBunVA,8C;QAWI,WpB5nVO,MAAO,KoB4nVG,cpB5nVH,EoB4kVH,KAi5BkB,OpB5nVf,C;QoB6nVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA06BI,MAAA,gE;MpB1oVA,iB;MoB0oVA,8C;QAWI,WpB/oVO,MAAO,KoB+oVG,cpB/oVH,EoB+lVH,KAo6BkB,OpB/oVf,C;QoBgpVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA07BI,MAAA,gE;MpB7pVA,iB;MoB6pVA,8C;QAWI,WpBlqVO,MAAO,KoBkqVG,cpBlqVH,EoBkmVH,KAu7BkB,OpBlqVf,C;QoBmqVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA08BI,MAAA,gE;MpBhrVA,iB;MoBgrVA,8C;QAWI,WpBrrVO,MAAO,KoBqrVG,cpBrrVH,EoBqnVH,KA08BkB,OpBrrVf,C;QoBsrVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA09BI,MAAA,gE;MpBnsVA,iB;MoBmsVA,8C;QAWI,WpBxsVO,MAAO,KoBwsVG,cpBxsVH,EoBwoVH,KA69BkB,OpBxsVf,C;QoBysVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0+BI,MAAA,gE;MpBttVA,iB;MoBstVA,8C;QAWI,WpB3tVO,MAAO,KoB2tVG,cpB3tVH,EoB2pVH,KAg/BkB,OpB3tVf,C;QoB4tVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0/BI,MAAA,gE;MpBzuVA,iB;MoByuVA,8C;QAWI,WpB9uVO,MAAO,KoB8uVG,cpB9uVH,EoB8qVH,KAmgCkB,OpB9uVf,C;QoB+uVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0gCI,MAAA,gE;MpB5vVA,iB;MoB4vVA,8C;QAWI,WpBjwVO,MAAO,KoBiwVG,cpBjwVH,EoBirVH,KAshCkB,OpBjwVf,C;QoBkwVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0hCI,MAAA,gE;MpB/wVA,iB;MoB+wVA,8C;QAWI,WpBpxVO,MAAO,KoBoxVG,cpBpxVH,EoBosVH,KAyiCkB,OpBpxVf,C;QoBqxVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0iCI,MAAA,gE;MpBlyVA,iB;MoBkyVA,8C;QAWI,WpBvyVO,MAAO,KoBuyVG,cpBvyVH,EoBuuVH,KA4jCkB,OpBvyVf,C;QoBwyVd,WAAW,eAAa,IAAb,C;QACX,aAAU,CAV,MAAkB,IAAlB,M;UACI,IAAK,WAAqB,GAAP,sBAAK,CAAL,CAAO,EAAnB,KAAqB,CAAM,CAAN,CAAF,CAAQ,C;;UAAT,OAAO,I;O;KAhBhC,C;+EAmBI,yB;MA0jCI,MAAA,gE;MpBrzVA,iB;MoBqzVA,8C;QAWI,WpB1zVO,MAAO,KoB0zVG,cpB1zV

OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,uBAAK,UAAL,EAAK,kBAAL,UAAV,EAAqB,OAArB,CAAJ,C;;QAET,OAAO,I;O;KAlBX,C;IAqBA,kC;MA2DI,WpBn3TO,MAAO,KoBm3TG,cpBn3TH,EoBi0TH,KAkDkB,KpBn3Tf,C;MoBo3Td,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WArDqB,GAqDP,sBAAK,CAAL,CArDO,EAAnB,KAqDqB,aAAM,CAAN,CArDF,CAqDqB,C;;MArDT,OAuDO,I;K;IApDX,kC;MAkEI,WpBt4TO,MAAO,KoBs4TG,cpBt4TH,EoB60TH,KAyDkB,KpBt4Tf,C;MoBu4Td,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA5DqB,GA4DP,sBAAK,CAAL,CA5DO,EAAnB,KA4DqB,aAAM,CAAN,CA5DF,CA4DqB,C;;MA5DT,OA8DO,I;K;IA3DX,kC;MAyEI,WpBz5TO,MAAO,KoBy5TG,cpBz5TH,EoBy1TH,KAgEkB,KpBz5Tf,C;MoB05Td,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAnEqB,GAmEP,sBAAK,CAAL,CAnEO,EAAnB,KAmEqB,aAAM,CAAN,CAnEF,CAmEqB,C;;MAnET,OAqEO,I;K;IAlEX,kC;MAgFI,WpB55TO,MAAO,KoB45TG,cpB55TH,EoB42TH,KAuEkB,KpB55Tf,C;MoB65Td,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA1EqB,GA0EP,sBAAK,CAAL,CA1EO,EAAnB,KA0EqB,aAAM,CAAN,CA1EF,CA0EqB,C;;MA1ET,OA4EO,I;K;IAzEX,kC;MAuFI,WpB/6TO,MAAO,KoB+6TG,cpB/6TH,EoB+3TH,KA8EkB,KpB/6Tf,C;MoBg7Td,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAjFqB,GAiFP,sBAAK,CAAL,CAjFO,EAAnB,KAiFqB,aAAM,CAAN,CAjFF,CAiFqB,C;;MAjFT,OAmFO,I;K;IAhFX,kC;MA8FI,WpBl8TO,MAAO,KoBk8TG,cpBl8TH,EoBk5TH,KAqFkB,KpBl8Tf,C;MoBm8Td,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAxFqB,GAwFP,sBAAK,CAAL,CAxFO,EAAnB,KAwFqB,aAAM,CAAN,CAxFF,CAwFqB,C;;MAxFT,OA0FO,I;K;IAvFX,kC;MAqGI,WpBr9TO,MAAO,KoBq9TG,cpBr9TH,EoBq6TH,KA4FkB,KpBr9Tf,C;MoBs9Td,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA/FqB,GA+FP,sBAAK,CAAL,CA/FO,EAAnB,KA+FqB,aAAM,CAAN,CA/FF,CA+FqB,C;;MA/FT,OAiGO,I;K;IA9FX,kC;MA4GI,WpBx+TO,MAAO,KoBw+TG,cpBx+TH,EoBw7TH,KAmGkB,KpBx+Tf,C;MoBy+Td,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAtGqB,GAsGP,sBAAK,CAAL,CAtGO,EAAnB,KAsGqB,aAAM,CAAN,CAtGF,CAsGqB,C;;MAtGT,OAwGO,I;K;IArGX,kC;MAmHI,WpB3/TO,MAAO,KoB2/TG,cpB3/TH,EoB28TH,KA0GkB,KpB3/Tf,C;MoB4/Td,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA7GqB,GA6GP,sBAAK,CAAL,CA7GO,EAAnB,KA6GqB,aAAM,CAAN,CA7GF,CA6GqB,C;;MA7GT,OA+GO,I;K;IA5GX,kC;MA0HI,WpB9gUO,MAAO,KoB8gUG,cpB9gUH,EoB89TH,KAiHkB,KpB9gUf,C;MoB+gUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WApHqB,GAoHP,sBAAK,CAAL,CApHO,EAAnB,KAoHqB,aAAM,CAAN,CApHF,CAoHqB,C;;MApHT,OAsHO,I;K;IAnHX,kC;MAiII,WpBjiUO,MAAO,KoBiiUG,cpBjiUH,EoBi/TH,KAwHkB,KpBjiUf,C;MoBkiUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA3HqB,GA2HP,sBAAK,CAAL,CA3HO,EAAnB,KA2HqB,aAAM,CAAN,CA3HF,CA2HqB,C;;MA3HT,OA6HO,I;K;IA1HX,kC;MAwII,WpBpjUO,MAAO,KoBojUG,cpBpjUH,EoBogUH,KA+HkB,KpBpjUf,C;MoBqjUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAlIqB,GAkIP,sBAAK,CAAL,CAlIO,EAAnB,KAkIqB,aAAM,CAAN,CAlIF,CAkIqB,C;;MAlIT,OAoIO,I;K;IAjIX,kC;MA+II,WpBvkUO,MAAO,KoBukUG,cpBvkUH,EoBuhUH,KAsIkB,KpBvkUf,C;MoBwkUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAzIqB,GAyIP,sBAAK,CAAL,CAzIO,EAAnB,KAyIqB,aAAM,CAAN,CAzIF,CAyIqB,C;;MAzIT,OA2IO,I;K;IAxIX,kC;MAsJI,WpB1lUO,MAAO,KoB0lUG,cpB1lUH,EoB0iUH,KA6IkB,KpB1lUf,C;MoB2lUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAhJqB,GAgJP,sBAAK,CAAL,CAhJO,EAAnB,KAgJqB,aAAM,CAAN,CAhJF,CAgJqB,C;;MAhJT,OAkJO,I;K;IA/IX,kC;MA6JI,WpB7mUO,MAAO,KoB6mUG,cpB7mUH,EoB6jUH,KAoJkB,KpB7mUf,C;MoB8mUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAvJqB,GAuJP,sBAAK,CAAL,CAvJO,EAAnB,KAuJqB,aAAM,CAAN,CAvJF,CAuJqB,C;;MAvJT,OAyJO,I;K;IAtJX,kC;MAoKI,WpBhoUO,MAAO,KoBgoUG,cpBhoUH,EoBgkUH,KA2JkB,KpBhoUf,C;MoBioUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA9JqB,GA8JP,sBAAK,CAAL,CA9JO,EAAnB,KA8JqB,aAAM,CAAN,CA9JF,CA8JqB,C;;MA9JT,OAgKO,I;K;IA7JX,kC;MA2KI,WpBnpUO,MAAO,KoBmpUG,cpBnpUH,EoBmlUH,KAkKkB,KpBnpUf,C;MoBopUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WArKqB,GAqKP,sBAAK,CAAL,CArKO,EAAnB,KAqKqB,aAAM,CAAN,CArKF,CAqKqB,C;;MArKT,OAuKO,I;K;IApKX,kC;MAkLI,WpBtqUO,MAAO,KoBsqUG,cpBtqUH,EoBsmUH,KAyKkB,KpBtqUf,C;MoBuqUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA5KqB,GA4KP,sBAAK,CAAL,CA5KO,EAAnB,KA4KqB,aAAM,CAAN,CA5KF,CA4KqB,C;;MA5KT,OA8KO,I;K;IA3KX,kC;MAyLI,WpBzrUO,MAAO,KoByrUG,cpBzrUH,EoBynUH,KAgLkB,KpBzrUf,C;MoB0rUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAnLqB,GAmLP,sBAAK,CAAL,CAnLO,EAAnB,KAmLqB,aAAM,CAAN,CAnLF,CAmLqB,C;;MAnLT,OAqLO,I;K;IAlLX,kC;MAgMI,WpB5sUO,MAAO,KoB4sUG,cpB5sUH,EoB4oUH,KAuLkB,KpB5sUf,C;MoB6sUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA1LqB,GA0LP,sBAAK,CAAL,CA1LO,EAAnB,KA0LqB,aAAM,CAAN,CA1LF,CA0LqB,C;;MA1LT,OA4LO,I;K;IAzLX,kC;MAuMI,WpB/tUO,MAAO,KoB+tUG,cpB/tUH,EoB+pUH,KA8LkB,KpB/tUf,C;MoBguUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAjMqB,GAiMP,sBAAK,CAAL,CAjMO,EAAnB,KAiMqB,aAAM,CAAN,CAjMF,CAiMqB,C;;MAjMT,OAmMO,I;K;IAhMX,kC;MA8MI,WpBlvUO,MAAO,KoBkvUG,cpBlvUH,EoBkrUH,KAqMkB,KpBlvUf,C;MoBmvUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAxMqB,GAwMP,sBAAK,CAAL,CAxMO,EAAnB,KAwMqB,aAAM,CAAN,CAxMF,CAwMqB,C;;MAxMT,OA0MO,I;K;IAvMX,kC;MAqNI,WpBrwUO,MAAO,KoBqwUG,cpBrwUH,EoBqsUH,KA4MkB,KpBrwUf,C;MoBswUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA/MqB,GA+MP,sBAAK,CAAL,CA/MO,EAAnB,KA+MqB,aAAM,CAAN,CA/MF,CA+MqB,C;;MA/MT,OAiNO,I;K;IA9MX,kC;MA4NI,WpBxxUO,MAAO,KoBwxUG,cpBxxUH,EoBwtUH,KAmNkB,KpBxxUf,C;MoByxUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAtNqB,GAsNP,sBAAK,CAAL,CAtNO,EAAnB,KAsNqB,aAAM,CAAN,CAtNF,CAsNqB,C;;MAtNT,OAwNO,I;K;IArNX,kC;MAmOI,WpB3yUO,MAAO,KoB2yUG,cpB3yUH,EoB2uUH,KA0NkB,KpB3yUf,C;MoB4yUd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA7NqB,GA6NP,sBAAK,CAAL,CA7NO,EAAnB,KA6NqB,aAAM,CAAN,CA7NF,CA6NqB,C;;MA7NT,OA+NO,I;K;IA5NX,kC;MA0OI,WpBj0UO,MAAO,KoBi0UG,cpBj0UH,EoBivUH,KAiOkB,KpBj0Uf,C;MoBk0Ud,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WApOqB,GAoOP,sBAAK,CAAL,CApOO,EAAnB,KAoOqB,aAAM,CAAN,CApOF,CAoOqB,C;;MApOT,OAsOO,I;K;IAnOX,kC;MAiPI,WpBv1UO,MAAO,KoBu1UG,cpBv1UH,EoBuwUH,KAwOkB,KpBv1Uf,C;MoBw1Ud,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA3OqB,GA2OP,sBAAK,CAAL,CA3OO,EAAnB,KA2OqB,aAAM,CAAN,CA3OF,CA2OqB,C;;MA3OT,OA6OO,I;K;IA1OX,kC;MAwPI,WpB72UO,MAAO,KoB62UG,cpB72UH,EoB6xUH,KA+OkB,KpB72Uf,C;MoB82Ud,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAlPqB,GAkPP,sBAAK,CAAL,CAlPO,EAAnB,KAkPqB,aAAM,CAAN,CAlPF,CAkPqB,C;;MAlPT,OAoPO,I;K;IAjPX,kC;MA+PI,WpBn4UO,MAAO,KoBm4UG,cpBn4UH,EoBmzUH,KAsPkB,KpBn4Uf,C;MoBo4Ud,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAzPqB,GAyPP,sBAAK,CAAL,CAzPO,EAAnB,KAyPqB,aAAM,CAAN,CAzPF,CAyPqB,C;;MAzPT,OA2PO,I;K;IAxPX,kC;MAsQI,WpBz5UO,MAAO,KoBy5UG,cpBz5UH,EoBy1UH,KA6PkB,KpBz5Uf,C;MoB05Ud,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAhQqB,GAgQP,sBAAK,CAAL,CAhQO,EAAnB,KAgQqB,aAAM,CAAN,CAhQF,CAgQqB,C;;MAhQT,OAkQO,I;K;IA/PX,kC;MA6QI,WpB76UO,MAAO,KoB66UG,cpB76UH,EoB61UH,KAoQkB,KpB76Uf,C;MoB86Ud,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAvQqB,GAuQP,sBAAK,CAAL,CAvQO,EAAnB,KAuQqB,aAAM,CAAN,CAvQF,CAuQqB,C;;MAvQT,OAyQO,I;K;IAtQX,kC;MAoRI,WpBh8UO,MAAO,KoBg8UG,cpBh8UH,EoBg3UH,KA2QkB,KpBh8Uf,C;MoBi8Ud,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA9QqB,GA8QP,sBAAK,CAAL,CA9QO,EAAnB,KA8QqB,aAAM,CAAN,CA9QF,CA8QqB,C;;MA9QT,OAgRO,I;K;IA7QX,kC;MA2RI,WpBn9UO,MAAO,KoBm9UG,cpBn9UH,EoBm4UH,KAkRkB,KpBn9Uf,C;MoBo9Ud,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WArRqB,GAqRP,sBAAK,CAAL,CArRO,EAAnB,KAqRqB,aAAM,CAAN,CArRF,CAqRqB,C;;MArRT,OAuRO,I;K;IApRX,kC;MAkSI,WpBt+UO,MAAO,KoBs+UG,cpBt+UH,EoBs5UH,KAyRkB,KpBt+Uf,C;MoBu+Ud,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA5RqB,GA4RP,sBAAK,CAAL,CA5RO,EAAnB,KA4RqB,aAAM,CAAN,CA5RF,CA4RqB,C;;MA5RT,OA8RO,I;K;IA3RX,kC;MAyST,WpBz/UO,MAAO,KoBy/UG,cpBz/UH,EoBy6UH,KAgSkB,KpBz/Uf,C;MoB0/Ud,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAnSqB,GAmSP,sBAAK,CAAL,CAnSO,EAAnB,KAmSqB,aAAM,CAAN,CAnSF,CAmSqB,C;;MAnST,OAqSO,I;K;IAlSX,kC;MAgTI,WpB5gVO,MAAO,KoB4gVG,cpB5gVH,EoB47UH,KAuSkB,KpB5gVf,C;MoB6gVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA1SqB,GA0SP,sBAAK,CAAL,CA1SO,EAAnB,KA0SqB,aAAM,CAAN,CA1SF,CA0SqB,C;;MA1ST,OA4SO,I;K;IAzSX,kC;MAuTI,WpB/hVO,MAAO,KoB+hVG,cpB/hVH,EoB+9UH,KA8SkB,KpB/hVf,C;MoBgiVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAjTqB,GAiTP,sBAAK,CAAL,CAjTO,EAAnB,KAiTqB,aAAM,CAAN,CAjTF,CAiTqB,C;;MAjTT,OAmTO,I;K;IAhTX,kC;MA8TI,WpBljVO,MAAO,KoBkjVG,cpBljVH,EoBkgVH,KAqTkB,KpBljVf,C;MoBmjVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAxTqB,GAwTP,sBAAK,CAAL,CAxTO,EAAnB,KAwTqB,aAAM,CAAN,CAxTF,CAwTqB,C;;MAxTT,OA0TO,I;K;IAvTX,kC;MAqUI,WpBrkVO,MAAO,KoBqkVG,cpBrkVH,EoBqhVH,KA4TkB,KpBrkVf,C;MoBskVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA/TqB,GA+TP,sBAAK,CAAL,CA/TO,EAAnB,KA+TqB,aAAM,CAAN,CA/TF,CA+TqB,C;;MA/TT,OAiUO,I;K;IA9TX,kC;MA4UI,WpBxlVO,MAAO,KoBwlVG,cpBxlVH,EoBwiVH,KAmUkB,KpBxlVf,C;MoBylVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAtUqB,GAsUP,sBAAK,CAAL,CAtUO,EAAnB,KAsUqB,aAAM,CAAN,CAtUF,CAsUqB,C;;MAtUT,OAwUO,I;K;IArUX,kC;MAmVI,WpB3mVO,MAAO,KoB2mVG,cpB3mVH,EoB2jVH,KA0UkB,KpB3mVf,C;MoB4mVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA7UqB,GA6UP,sBAAK,CAAL,CA7UO,EAAnB,KA6UqB,aAAM,CAAN,CA7UF,CA6UqB,C;;MA7UT,OA+UO,I;K;IA5UX,kC;MA0VI,WpBjoVO,MAAO,KoBioVG,cpBjoVH,EoBimVH,KAiVkB,KpBjoVf,C;MoBkoVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WApVqB,GAoVP,sBAAK,CAAL,CApVO,EAAnB,KAoVqB,aAAM,CAAN,CApVF,CAoVqB,C;;MApVT,OAsVO,I;K;IAnVX,kC;MAiWI,WpBvpVO,MAAO,KoBupVG,cpBvpVH,EoBunVH,KAwVkB,KpBvpVf,C;MoBwpVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA3VqB,GA2VP,sBAAK,CAAL,CA3VO,EAAnB,KA2VqB,aAAM,CAAN,CA3VF,CA2VqB,C;;MA3VT,OA6VO,I;K;IA1VX,kC;MAwWI,WpB7qVO,MAAO,KoB6qVG,cpB7qVH,EoB6oVH,KA+VkB,KpB7qVf,C;MoB8qVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAlWqB,GAkWP,sBAAK,CAAL,CAlWO,EAAnB,KAkWqB,aAAM,CAAN,CAlWF,CAkWqB,C;;MAlWT,OAoWO,I;K;IAjWX,kC;MA+WI,WpBnsVO,MAAO,KoBmsVG,cpBnsVH,EoBmqVH,KAsWkB,KpBnsVf,C;MoBosVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAzWqB,GAyWP,sBAAK,CAAL,CAzWO,EAAnB,KAyWqB,aAAM,CAAN,CAzWF,CAyWqB,C;;MAzWT,OA2WO,I;K;IAxWX,kC;MAsXI,WpBztVO,MAAO,KoBytVG,cpBztVH,EoBysrVH,KA6WkB,KpBztVf,C;MoB0tVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAhXqB,GAgXP,sBAAK,CAAL,CAhXO,EAAnB,KAgXqB,aAAM,CAAN,CAhXF,CAgXqB,C;;MAhXT,OAkXO,I;K;IA/WX,kC;MA6XI,WpBjvVO,MAAO,KoBivVG,cpBjvVH,EoBiurVH,KAoXkB,KpBjvVf,C;MoBkvVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAvXqB,GAuXP,sBAAK,CAAL,CAvXO,EAAnB,KAuXqB,aAAM,CAAN,CAvXF,CAuXqB,C;;MAvXT,OAyXO,I;K;IAtXX,kC;MAoYI,WpBhwVO,MAAO,KoBgwVG,cpBhwVH,EoBgvrVH,KA2XkB,KpBhwVf,C;MoBiwVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA9XqB,GA8XP,sBAAK,CAAL,CA9XO,EAAnB,KA8XqB,aAAM,CAAN,CA9XF,CA8XqB,C;;MA9XT,OAgYO,I;K;IA7XX,kC;MA2YI,WpBnxVO,MAAO,KoBmxVG,cpBnxVH,EoBmwrVH,KAkYkB,KpBnxVf,C;MoBoxVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WArYqB,GAqYP,sBAAK,CAAL,CArYO,EAAnB,KAqYqB,aAAM,CAAN,CArYF,CAqYqB,C;;MArYT,OAuYO,I;K;IApYX,kC;MAkZI,WpBtyVO,MAAO,KoBsyVG,cpBtyVH,EoBsxrVH,KAyYkB,KpBtyVf,C;MoBuyVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA5YqB,GA4YP,sBAAK,CAAL,CA5YO,EAAnB,KA4YqB,aAAM,CAAN,CA5YF,CA4YqB,C;;MA5YT,OA8YO,I;K;IA3YX,kC;MAyZI,WpBzzVO,MAAO,KoByzVG,cpBzzVH,EoBysrVH,KAgZkB,KpBzzVf,C;MoB0zVd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAnZqB,GAmZP,sBAAK,CAAL,CAnZO,EAAnB,KAmZqB,aAAM,CAAN,CAnZF,CAmZqB,C;;MAnZT,OAqZO,I;K;IAlZX,kC;MAgaI,WpB50VO,MAAO,KoB40VG,cpB50VH,EoB4xrVH,KAuZkB,KpB50Vf,C;MoB60Vd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAV,MAAkB,IAAlB,M;QACI,IAAK,WA1ZqB,GA0ZP,sBAAK,CAAL,CA1ZO,EAAnB,KA0ZqB,aAAM,CAAN,CA1ZF,CA0ZqB,C;;MA1ZT,OA4ZO,I;K;IAzZX,kC;MAuaI,WpB/1VO,MAAO,KoB+1VG,cpB/1VH,EoB+yrVH,KA8ZkB,KpB/1Vf,C;MoBg2Vd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAjaqB,GAiaP,sBAAK,CAAL,CAjaO,EAAnB,KAiaqB,aAAM,CAAN,CAjaF,CAiaqB,C;;MAjaT,OAmaO,I;K;IAhaX,kC;MA8aI,WpBl3VO,MAAO,KoBk3VG,cpBl3VH,EoBkzrVH,KAqakB,KpBl3Vf,C;MoBm3Vd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAxaqB,GAwaP,sBAAK,CAAL,CAxaO,EAAnB,KAwaqB,aAAM,CAAN,CAxaF,CAwaqB,C;;MAxaT,OA0aO,I;K;IAvaX,kC;MAqbI,WpBr4VO,MAAO,KoBq4VG,cpBr4VH,EoBq0rVH,KA4akB,KpBr4Vf,C;MoBs4Vd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA/aqB,GA+aP,sBAAK,CAAL,CA/aO,EAAnB,KA+aqB,aAAM,CAAN,CA/aF,CA+aqB,C;;MA/aT,OAibO,I;K;IA9aX,kC;MA4bI,WpBx5VO,MAAO,KoBw5VG,cpBx5VH,EoBw1rVH,KAmbkB,KpBx5Vf,C;MoBy5Vd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAtbqB,GAsbP,sBAAK,CAAL,CAtbO,EAAnB,KAsbqB,aAAM,CAAN,CAtbF,CAsbqB,C;;MAtbT,OAwbO,I;K;IArbX,kC;MAmcI,WpB36VO,MAAO,KoB26VG,cpB36VH,EoB23rVH,KA0bkB,KpB36Vf,C;MoB46Vd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA7bqB,GA6bP,sBAAK,CAAL,CA7bO,EAAnB,KA6bqB,aAAM,CAAN,CA7bF,CA6bqB,C;;MA7bT,OA+bO,I;K;IA5bX,kC;MA0cI,WpBj8VO,MAAO,KoBi8VG,cpBj8VH,EoBi5rVH,KAickB,KpBj8Vf,C;MoBk8Vd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WApcqB,GAocP,sBAAK,CAAL,CApcO,EAAnB,KAocqB,aAAM,CAAN,CApcF,CAocqB,C;;MApcT,OAscO,I;K;IAncX,kC;MAidI,WpBv9VO,MAAO,KoBu9VG,cpBv9VH,EoBu6rVH,KAwckB,KpBv9Vf,C;MoBw9Vd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA3cqB,GA2cP,sBAAK,CAAL,CA3cO,EAAnB,KA2cqB,aAAM,CAAN,CA3cF,CA2cqB,C;;MA3cT,OA6cO,I;K;IA1cX,kC;MAwdI,WpB7+VO,MAAO,KoB6+VG,cpB7+VH,EoB67rVH,KA+ckB,KpB7+Vf,C;MoB8+Vd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAldqB,GAkdP,sBAAK,CAAL,CAldO,EAAnB,KAkdqB,aAAM,CAAN,CAldF,CAkdqB,C;;MAldT,OAodO,I;K;IAjdX,kC;MA+dI,WpBngWO,MAAO,KoBmgWG,cpBngWH,EoBm9rVH,KAsdkB,KpBngWf,C;MoBogWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAzdqB,GAydP,sBAAK,CAAL,CAzdO,EAAnB,KAydqB,aAAM,CAAN,CAzdF,CAydqB,C;;MAzdT,OA2dO,I;K;IAxdX,kC;MAseI,WpBzhWO,MAAO,KoByhWG,cpBzhWH,EoBy+rVH,KA6dkB,KpBzhWf,C;MoB0hWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAheqB,GAgeP,sBAAK,CAAL,CAheO,EAAnB,KAgeqB,aAAM,CAAN,CAheF,CAgeqB,C;;MAheT,OAkeO,I;K;IA/dX,kC;MA6fI,WpBjjWO,MAAO,KoBijWG,cpBjjWH,EoBigsVH,KAoekB,KpBjjWf,C;MoBkjWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAvfqB,GAufP,sBAAK,CAAL,CAvfO,EAAnB,KAufqB,aAAM,CAAN,CAvfF,CAufqB,C;;MAvfT,OAyfO,I;K;IAtfX,kC;MAogBI,WpBrkWO,MAAO,KoBqkWG,cpBrkWH,EoBqhsVH,KA2fkB,KpBrkWf,C;MoBskWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA9fqB,GA8fP,sBAAK,CAAL,CA9fO,EAAnB,KA8fqB,aAAM,CAAN,CA9fF,CA8fqB,C;;MA9fT,OAggBO,I;K;IA7fX,kC;MA2hBI,WpBnmWO,MAAO,KoBmmWG,cpBnmWH,EoBmjsVH,KAkgBkB,KpBnmWf,C;MoBomWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WArgBqB,GAqgBP,sBAAK,CAAL,CArgBO,EAAnB,KAqgBqB,aAAM,CAAN,CArgBF,CAqgBqB,C;;MArgBT,OAugBO,I;K;IApgBX,kC;MAkiBI,WpBtnWO,MAAO,KoBsnWG,cpBtnWH,EoBsksVH,KAyhBkB,KpBtnWf,C;MoBunWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA5hBqB,GA4hBP,sBAAK,CAAL,CA5hBO,EAAnB,KA4hBqB,aAAM,CAAN,CA5hBF,CA4hBqB,C;;MA5hBT,OA8hBO,I;K;IA3hBX,kC;MAyiBI,WpBzoWO,MAAO,KoByoWG,cpBzoWH,EoBylsVH,KAgiBkB,KpBzoWf,C;MoB0oWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAniBqB,GAmiBP,sBAAK,CAAL,CAniBO,EAAnB,KAmiBqB,aAAM,CAAN,CAniBF,CAmiBqB,C;;MAniBT,OAqiBO,I;K;IAliBX,kC;MAgjBI,WpB5pWO,MAAO,KoB4pWG,cpB5pWH,EoB4msVH,KAuiBkB,KpB5pWf,C;MoB6pWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA1iBqB,GA0iBP,sBAAK,CAAL,CA1iBO,EAAnB,KA0iBqB,aAAM,CAAN,CA1iBF,CA0iBqB,C;;MA1iBT,OA4iBO,I;K;IAziBX,kC;MAujBI,WpB/qWO,MAAO,KoB+qWG,cpB/qWH,EoB+nsVH,KA8iBkB,KpB/qWf,C;MoBgrWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAjjBqB,GAijBP,sBAAK,CAAL,CAjjBO,EAAnB,KAijBqB,aAAM,CAAN,CAjjBF,CAijBqB,C;;MAjjBT,OAmjBO,I;K;IAhjBX,kC;MA8jBI,WpBlsWO,MAAO,KoBksWG,cpBlsWH,EoBkosVH,KAqjBkB,KpBlsWf,C;MoBmsWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAxjBqB,GAwjBP,sBAAK,CAAL,CAxjBO,EAAnB,KAwjBqB,aAAM,CAAN,CAxjBF,CAwjBqB,C;;MAxjBT,OA0jBO,I;K;IAvjBX,kC;MAqkBI,WpBrtWO,MAAO,KoBqtWG,cpBrtWH,EoBqpsVH,KA4jBkB,KpBrtWf,C;MoBstWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA/jBqB,GA+jBP,sBAAK,CAAL,CA/jBO,EAAnB,KA+jBqB,aAAM,CAAN,CA/jBF,CA+jBqB,C;;MA/jBT,OAikBO,I;K;IA9jBX,kC;MA4lBI,WpBxuWO,MAAO,KoBwuWG,cpBxuWH,EoBwqsVH,KAmlBkB,KpBxuWf,C;MoByuWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WAtlBqB,GAslBP,sBAAK,CAAL,CAtlBO,EAAnB,KAslBqB,aAAM,CAAN,CAtlBF,CAslBqB,C;;MAtlBT,OAwlBO,I;K;IArlBX,kC;MAmmBI,WpB3vWO,MAAO,KoB2vWG,cpB3vWH,EoB2rsVH,KA0lBkB,KpB3vWf,C;MoB4vWd,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAlB,M;QACI,IAAK,WA7lBqB,GA6lBP,sBAAK,CAAL,CA7lBO,EAAnB,KA6lBqB,aAAM,CAAN,CA7lBF,CA6lBqB,C;;MA7lBT,OA+lBO,I;K;IA5

AAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,oBAAO,cAAP,EAAO,sBAAP,WAAkB,OAAlB,C;;MACJ,OAAO,M;K;IAGX,oC;MAQoB,UACL,M;MAHX,aAAa,iBAAY,cAAZ,C;MACb,YAAY,C;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,oBAAO,cAAP,EAAO,sBAAP,WAAkB,OAAlB,C;;MACJ,OAAO,M;K;IAGX,2B;MAQoB,Q;MADhB,UAAgB,W;MACA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MpCAiD,SoCAjD,GpCA2D,KAAK,GoCAzD,OpCAoE,KAAX,IAAf,C;;MoCErD,OAAO,G;K;IAGX,2B;MAQoB,Q;MADhB,UAAiB,2B;MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MpBXmD,UoBWnD,GpBX8D,KAAK,KoBW5D,OpBXuE,KAAX,CAAhB,C;;MoBavD,OAAO,G;K;IAGX,2B;MAQoB,Q;MADhB,UAAgB,W;MACA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MpC5BiD,SoC4BjD,GpC5B2D,KAAK,GAAW,CD2O5C,SqC/MxB,OrC+MkC,KAAL,GAAiB,GAAtB,CC3O4C,MAAX,IAAf,C;;MoC8BrD,OAAO,G;K;IAGX,2B;MAQoB,Q;MADhB,UAAgB,W;MACA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MpC1CiD,SoC0CjD,GpC1C2D,KAAK,GAAW,CC4O5C,SmClMxB,OnCkMkC,KAAL,GAAiB,KAAtB,CD5O4C,MAAX,IAAf,C;;MoC4CrD,OAAO,G;K;IC3GX,wB;MAMI,OrCuCkE,SqCvClE,GrCuC4E,KqCvC5E,GrCuCsF,KAA7B,CqCvCzD,SqCuCmE,KAAK,GAAW,CqCvCnF,SrCuC6F,KAAK,GAAW,CqCvC7G,SrCuCuH,KAAK,GAAW,CqCvCvI,SrCuCiJ,KAAK,GAAW,CqCvCjK,SrCuC2K,KAAK,GAAW,CqCvC3L,SrCuCqM,KAAK,GAAW,CqCvCrN,SrCuC+N,KAAK,GAAW,CqCvC/O,SrCuCyP,KAAK,GAAW,CqCvCzQ,SrCuCmR,KAAK,GAAW,CqCvCnS,SrCuC6S,KAAK,GAAW,CqCvC7T,SrCuCuU,KAAK,GAAW,CqCvCvV,SrCuCiW,KAAK,GAAW,CqCvCjX,SrCuC2X,KAAK,GAAW,CqCvC3Y,SrCuCqZ,KAAK,GAAW,CqCvCra,SrCuC+a,KAAK,GAAW,CqCvC/b,SrCuCyc,KAAK,GAAW,CqCvCzd,SrCuCme,KAAK,GAAW,CqCvCne,SrCuC6e,KAAK,GAAW,CqCvC7f,SrCuCugB,KAAK,GAAW,CqCvCvhB,SrCuCiiB,KAAK,GAAW,CqCvCjjB,SrCuC2jB,KAAK,GAAW,CqCvC3kB,SrCuCqlB,KAAK,GAAW,CqCvCrmB,SrCuC+mB,KAAK,GAAW,CqCvC/nB,SrCuCyoB,KAAK,GAAW,CqCvCzpB,SrCuCmqB,KAAK,GAAW,CqCvCnrB,SrCuC6rB,KAAK,GAAW,CqCvC7sB,SrCuCutB,KAAK,GAAW,CqCvCvuB,SrCuCivB,KAAK,GAAW,CqCvCjwB,SrCuC2wB,KAAK,GAAW,CqCvC3xB,SrCuCqyB,KAAK,GAAW,CqCvCryB,SrCuC+yB,KAAK,GAAW,CqCvC/zB,SrCuCy0B,KAAK,GAAW,CqCvCz1B,SrCuCm2B,KAAK,GAAW,CqCvCn3B,SrCuC63B,KAAK,GAAW,CqCvC73B,SrCuCu4B,KAAK,GAAW,CqCvCv5B,SrCuCi6B,KAAK,GAAW,CqCvCj7B,SrCuC27B,KAAK,GAAW,CqCvC37B,SrCuCq8B,KAAK,GAAW,CqCvCr9B,SrCuC+9B,KAAK,GAAW,CqCvC/+B,SrCuCy/B,KAAK,GAAW,CqCvCz/B,SrCuCmgC,KAAK,GAAW,CqCvCngC,SrCuC6gC,KAAK,GAAW,CqCvC7hC,SrCuCuiC,KAAK,GAAW,CqCvCviC,SrCuCijC,KAAK,GAAW,CqCvCjkC,SrCuC2kC,KAAK,GAAW,CqCvC3lC,SrCuCqmC,KAAK,GAAW,CqCvCrnC,SrCuC+nC,KAAK,GAAW,CqCvC/oC,SrCuCypC,KAAK,GAAW,CqCvCzqC,SrCuCmrC,KAAK,GAAW,CqCvCnsC,SrCuC6sC,KAAK,GAAW,CqCvC7tC,SrCuCuuC,KAAK,GAAW,CqCvCvvC,SrCuCiwC,KAAK,GAAW,CqCvCjxC,SrCuC2xC,KAAK,GAAW,CqCvC3yC,SrCuCqzC,KAAK,GAAW,CqCvCr0C,SrCuC+0C,KAAK,GAAW,CqCvC/1C,SrCuCy2C,KAAK,GAAW,CqCvCz3C,SrCuCm4C,KAAK,GAAW,CqCvCn5C,SrCuC65C,KAAK,GAAW,CqCvC75C,SrCuCu6C,KAAK,GAAW,CqCvCv7C,SrCuCi8C,KAAK,GAAW,CqCvCj9C,SrCuC29C,KAAK,GAAW,CqCvC39C,SrCuCq+C,KAAK,GAAW,CqCvCr/C,SrCuC+/C,KAAK,GAAW,CqCvC//C,SrCuCygD,KAAK,GAAW,CqCvCzhD,SrCuCmiD,KAAK,GAAW,CqCvCniD,SrCuC6iD,KAAK,GAAW,CqCvC7jD,SrCuCukD,KAAK,GAAW,CqCvCvlD,SrCuCimD,KAAK,GAAW,CqCvCjnD,SrCuC2nD,KAAK,GAAW,CqCvC3oD,SrCuCqpD,KAAK,GAAW,CqCvCrqD,SrCuC+qD,KAAK,GAAW,CqCvC/rD,SrCuCysD,KAAK,GAAW,CqCvCztD,SrCuCmuD,KAAK,GAAW,CqCvCnvD,SrCuC6vD,KAAK,GAAW,CqCvC7wD,SrCuCuxD,KAAK,GAAW,CqCvCvyD,SrCuCizD,KAAK,GAAW,CqCvCjzD,SrCuC20D,KAAK,GAAW,CqCvC30D,SrCuCq1D,KAAK,GAAW,CqCvCr2D,SrCuC+2D,KAAK,GAAW,CqCvC/3D,SrCuCy4D,KAAK,GAAW,CqCvCz5D,SrCuCm6D,KAAK,GAAW,CqCvCn7D,SrCuC67D,KAAK,GAAW,CqCvC77D,SrCuCu8D,KAAK,GAAW,CqCvCv9D,SrCuCi+D,KAAK,GAAW,CqCvCj/D,SrCuC2/D,KAAK,GAAW,CqCvC3gE,SrCuCqhE,KAAK,GAAW,CqCvCrhE,SrCuC+hE,KAAK,GAAW,CqCvC/iE,SrCuCyjE,KAAK,GAAW,CqCvCzkE,SrCuCmlE,KAAK,GAAW,CqCvCnmE,SrCuC6mE,KAAK,GAAW,CqCvC7nE,SrCuCuoE,KAAK,GAAW,CqCvCvpE,SrCuCiqE,KAAK,GAAW,CqCvCjrE,SrCuC2rE,KAAK,GAAW,CqCvC3sE,SrCuCqtE,KAAK,GAAW,CqCvCrtE,SrCuC+tE,KAAK,GAAW,CqCvC/uE,SrCuCyvE,KAAK,GAAW,CqCvCzwE,SrCuCmxE,KAAK,GAAW,CqCvCnyE,SrCuC6yE,KAAK,GAAW,CqCvC7zE,SrCuCu0E,KAAK,GAAW,CqCvCv1E,SrCuCi2E,KAAK,GAAW,CqCvCj3E,SrCuC23E,KAAK,GAAW,CqCvC33E,SrCuCq4E,KAAK,GAAW,CqCvCr5E,SrCuC+5E,KAAK,GAAW,CqCvC/6E,SrCuCy7E,KAAK,GAAW,CqCvCz8E,SrCuCm9E,KAAK,GAAW,CqCvCn+E,SrCuC6+E,KAAK,GAAW,CqCvC7/E,SrCuCugF,KAAK,GAAW,CqCvCvhF,SrCuCiiF,KAAK,GAAW,CqCvCjjF,SrCuC2jF,KAAK,GAAW,CqCvC3kF,SrCuCqlF,KAAK,GAAW,CqCvCrmF,SrCuC+mF,KAAK,GAAW,CqCvC/nF,SrCuCyoF,KAAK,GAAW,CqCvCzpF,SrCuCmqF,KAAK,GAAW,CqCvCnrF,SrCuC6rF,KAAK,GAAW,CqCvC7sF,SrCuCutF,KAAK,GAAW,CqCvCvuF,SrCuCivF,KAAK,GAAW,CqCvCjwF,SrCuC2wF,KAAK,GAAW,CqCvC3xF,SrCuCqyF,KAAK,GAAW,CqCvCryF,SrCuC+yF,KAAK,GAAW,CqCvC/zF,SrCuCy0F,KAAK,GAAW,CqCvCz1F,SrCuCm2F,KAAK,GAAW,CqCvCn3F,SrCuC63F,KAAK,GAAW,CqCvC73F,SrCuCu4F,KAAK,GAAW,CqCvCv5F,SrCuCi6F,KAAK,GAAW,CqCvCj7F,SrCuC27F,KAAK,GAAW,CqCvC37F,SrCuCq8F,KAAK,GAAW,CqCvCr9F,SrCuC+9F,KAAK,GAAW,CqCvC/+F,SrCuCy/F,KAAK,GAAW,CqCvCz/F,SrCuCmgG,KAAK,GAAW,CqCvCngG,SrCuC6gG,KAAK,GAAW,CqCvC7hG,SrCuCuiG,KAAK,GAAW,CqCvCviG,SrCuCijG,KAAK,GAAW,CqCvCjkG,SrCuC2kG,KAAK,GAAW,CqCvC3lG,SrCuCqmG,KAAK,GAAW,CqCvCrnG,SrCuC+nG,KAAK,GAAW,CqCvC/oG,SrCuCypG,KAAK,GAAW,CqCvCzqG,SrCuCmrG,KAAK,GAAW,CqCvCnsG,SrCuC6sG,KAAK,GAAW,CqCvC7tG,SrCuCuuG,KAAK,GAAW,CqCvCvvG,SrCuCiwG,KAAK,GAAW,CqCvCjxG,SrCuC2xG,KAAK,GAAW,CqCvC3yG,SrCuCqzG,KAAK,GAAW,CqCvCr0G,SrCuC+0G,KAAK,GAAW,CqCvC/1G,SrCuCy2G,KAAK,GAAW,CqCvCz3G,SrCuCm4G,KAAK,GAAW,CqCvCn5G,SrCuC65G,KAAK,GAAW,CqCvC75G,SrCuCu6G,KAAK,GAAW,CqCvCv7G,SrCuCi8G,KAAK,GAAW,CqCvCj9G,SrCuC29G,KAAK,GAAW,CqCvC39G,SrCuCq+G,KAAK,GAAW,CqCvCr/G,SrCuC+/G,KAAK,GAAW,CqCvC//G,SrCuCygH,KAAK,GAAW,CqCvCzhH,SrCuCmiH,KAAK,GAAW,CqCvCniH,SrCuC6iH,KAAK,GAAW,CqCvC7jH,SrCuCukH,KAAK,GAAW,CqCvCvlH,SrCuCimH,KAAK,GAAW,CqCvCjnH,SrCuC2nH,KAAK,GAAW,CqCvC3oH,SrCuCqpH,KAAK,GAAW,CqCvCrqH,SrCuC+qH,KAAK,GAAW,CqCvC/rH,SrCuCysH,KAAK,GAAW,CqCvCztH,SrCuCmuH,KAAK,GAAW,CqCvCnvH,SrCuC6vH,KAAK,GAAW,CqCvC7wH,SrCuCuxH,KAAK,GAAW,CqCvCvyH,SrCuCizH,KAAK,GAAW,CqCvCjzH,SrCuC20H,KAAK,GAAW,CqCvC30H,SrCuCq1H,KAAK,GAAW,CqCvCr2H,SrCuC+2H,KAAK,GAAW,CqCvC/3H,SrCuCy4H,KAAK,GAAW,CqCvCz5H,SrCuCm6H,KAAK,GAAW,CqCvCn7H,SrCuC67H,KAAK,GAAW,CqCvC77H,SrCuCu8H,KAAK,GAAW,CqCvCv9H,SrCuCi+H,KAAK,GAAW,CqCvCj/H,SrCuC2/H,KAAK,GAAW,CqCvC3gI,SrCuCqhI,KAAK,GAAW,CqCvCrhI,SrCuC+hI,KAAK,GAAW,CqCvC/iI,SrCuCyjI,KAAK,GAAW,CqCvCzkI,SrCuCmlI,KAAK,GAAW,CqCvCnmI,SrCuC6mI,KAAK,GAAW,CqCvC7nI,SrCuCuoI,KAAK,GAAW,CqCvCvpI,SrCuCiqI,KAAK,GAAW,CqCvCjrI,SrCuC2rI,KAAK,GAAW,CqCvC3sI,SrCuCqtI,KAAK,GAAW,CqCvCrtI,SrCuC+tI,KAAK,GAAW,CqCvC/uI,SrCuCyvI,KAAK,GAAW,CqCvCzwI,SrCuCmxI,KAAK,GAAW,CqCvCnyI,SrCuC6yI,KAAK,GAAW,CqCvC7zI,SrCuCu0I,KAAK,GAAW,CqCvCv1I,SrCuCi2I,KAAK,GAAW,CqCvCj3I,SrCuC23I,KAAK,GAAW,CqCvC33I,SrCuCq4I,KAAK,GAAW,CqCvCr5I,SrCuC+5I,KAAK,GAAW,CqCvC/6I,SrCuCy7I,KAAK,GAAW,CqCvCz8I,SrCuCm9I,KAAK,GAAW,CqCvCn+I,SrCuC6+I,KAAK,GAAW,CqCvC7/I,SrCuCugJ,KAAK,GAAW,CqCvCvhJ,SrCuCiiJ,KAAK,GAAW,CqCvCjjJ,SrCuC2jJ,KAAK,GAAW,CqCvC3kJ,SrCuCqlJ,KAAK,GAAW,CqCvCrmJ,SrCuC+mJ,KAAK,GAAW,CqCvC/nJ,SrCuCyoJ,KAAK,GAAW,CqCvCzpJ,SrCuCmqJ,KAAK,GAAW,CqCvCnrJ,SrCuC6rJ,KAAK,GAAW,CqCvC7sJ,SrCuCutJ,KAAK,GAAW,CqCvCvuJ,SrCuCivJ,KAAK,GAAW,CqCvCjwJ,SrCuC2wJ,KAAK,GAAW,CqCvC3xJ,SrCuCqyJ,KAAK,GAAW,CqCvCryJ,SrCuC+yJ,KAAK,GAAW,CqCvC/zJ,SrCuCy0J,KAAK,GAAW,CqCvCz1J,SrCuCm2J,KAAK,GAAW,CqCvCn3J,SrCuC63J,KAAK,GAAW,CqCvC73J,SrCuCu4J,KAAK,GAAW,CqCvCv5J,SrCuCi6J,KAAK,GAAW,CqCvCj7J,SrCuC27J,KAAK,GAAW,CqCvC37J,SrCuCq8J,KAAK,GAAW,CqCvCr9J,SrCuC+9J,KAAK,GAAW,CqCvC/+J,SrCuCy/J,KAAK,GAAW,CqCvCz/J,SrCuCmgK,KAAK,GAAW,CqCvCngK,SrCuC6gK,KAAK,GAAW,CqCvC7hK,SrCuCuiK,KAAK,GAAW,CqCvCviK,SrCuCijK,KAAK,GAAW,CqCvCjkK,SrCuC2kK,KAAK,GAAW,CqCvC3lK,SrCuCqmK,KAAK,GAAW,CqCvCrnK,SrCuC+nK,KAAK,GAAW,CqCvC/oK,SrCuCypK,KAAK,GAAW,CqCvCzqK,SrCuCmrK,KAAK,GAAW,CqCvCnsK,SrCuC6sK,KAAK,GAAW,CqCvC7tK,SrCuCuuK,KAAK,GAAW,CqCvCvvK,SrCuCiwK,KAAK,GAAW,CqCvCjxK,SrCuC2xK,KAAK,GAAW,CqCvC3yK,SrCuCqzK,KAAK,GAAW,CqCvCr0K,SrCuC+0K,KAAK,GAAW,CqCvC/1K,SrCuCy2K,KAAK,GAAW,CqCvCz3K,SrCuCm4K,KAAK,GAAW,CqCvCn5K,SrCuC65K,KAAK,GAAW,CqCvC75K,SrCuCu6K,KAAK,GAAW,CqCvCv7K,SrCuCi8K,KAAK,GAAW,CqCvCj9K,SrCuC29K,KAAK,GAAW,CqCvC39K,SrCuCq+K,KAAK,GAAW,CqCvCr/K,SrCuC+/K,KAAK,GAAW,CqCvC//K,SrCuCygL,KAAK,GAAW,CqCvCzhL,SrCuCmiL,KAAK,GAAW,CqCvCniL,SrCuC6iL,KAAK,GAAW,CqCvC7jL,SrCuCukL,KAAK,GAAW,CqCvCvlL,SrCuCimL,KAAK,GAAW,CqCvCjnL,SrCuC2nL,KAAK,GAAW,CqCvC3oL,SrCuCqpL,KAAK,GAAW,CqCvCrqL,SrCuC+qL,KAAK,GAAW,CqCvC/rL,SrCuCysL,KAAK,GAAW,CqCvCztL,SrCuCmuL,KAAK,GAAW,CqCvCnvL,SrCuC6vL,KAAK,GAAW,CqCvC7wL,SrCuCuxL,KAAK,GAAW,CqCvCvyL,SrCuCizL,KAAK,GAAW,CqCvCjzL,SrCuC20L,KAAK,GAAW,CqCvC30L,SrCuCq1L,KAAK,GAAW,CqCvCr2L,SrCuC+2L,KAAK,GAAW,CqCvC/3L,SrCuCy4L,KAAK,GAAW,CqCvCz5L,SrCuCm6L,KAAK,GAAW,CqCvCn7L,SrCuC67L,KAAK,GAAW,CqCvC77L,SrCuCu8L,KAAK,GAAW,CqCvCv9L,SrCuCi+L,KAAK,GAAW,CqCvCj/L,SrCuC2/L,KAAK,GAAW,CqCvC3gM,SrCuCqhM,KAAK,GAAW,CqCvCrhM,SrCuC+hM,KAAK,GAAW,CqCvC/iM,SrCuCyjM,KAAK,GAAW,CqCvCzkM,SrCuCmlM,KAAK,GAAW,CqCvCnmM,SrCuC6mM,KAAK,GAAW,CqCvC7nM,SrCuCuoM,KAAK,GAAW,CqCvCvpM,SrCuCiqM,KAAK,GAAW,CqCvCjrM,SrCuC2rM,KAAK,GAAW,CqCvC3sM,SrCuCqtM,KAAK,GAAW,CqCvCrtM,SrCuC+tM,KAAK,GAAW,CqCvC/uM,SrCuCyvM,KAAK,GAAW,CqCvCzwM,SrCuCmxM,KAAK,GAAW,CqCvCnyM,SrCuC6yM,KAAK,GAAW,CqCvC7zM,SrCuCu0M,KAAK,GAAW,CqCvCv1M,SrCuCi2M,KAAK,GAAW,CqCvCj3M,SrCuC23M,KAAK,GAAW,CqCvC33M,SrCuCq4M,KAAK,GAAW,CqCvCr5M,SrCuC+5M,KAAK,GAAW,CqCvC/6M,SrCuCy7M,KAAK,GAAW,CqCvCz8M,SrCuCm9M,KAAK,GAAW,CqCvCn+M,SrCuC6+M,KAAK,GAAW,CqCvC7/M,SrCuCugN,KAAK,GAAW,CqCvCvhN,SrCuCiiN,KAAK,GAAW,CqCvCjjN,SrCuC2jN,KAAK,GAAW,CqCvC3kN,SrCuCqlN,KAAK,GAAW,CqCvCrmN,SrCuC+mN,KAAK,GAAW,CqCvC/nN,SrCuCyoN,KAAK,GAAW,CqCvCzpN,SrCuCmqN,KAAK,GAAW,CqCvCnrN,SrCuC6rN,KAAK,GAAW,CqCvC7sN,SrCuCutN,KAAK,GAAW,CqCvCvuN,SrCuCivN,KAAK,GAAW,CqCvCjwN,SrCuC2wN,KAAK,GAAW,CqCvC3xN,SrCuCqyN,KAAK,GAAW,CqCvCryN,SrCuC+yN,KAAK,GAAW,CqCvC/zN,SrCuCy0N,KAAK,GAAW,CqCvCz1N,SrCuCm2N,KAAK,GAAW,CqCvCn3N,SrCuC63N,KAAK,GAAW,CqCvC73N,SrCuCu4N,KAAK,GAAW,CqCvCv5N,SrCuCi6N,KAAK,GAAW,CqCvCj7N,SrCuC27N,KAAK,GAAW,CqCvC37N,SrCuCq8N,KAAK,GAAW,CqCvCr9N,SrCuC+9N,KAAK,GAAW,CqCvC/+N,SrCuCy/N,KAAK,GAAW,CqCvCz/N,SrCuCmgO,KAAK,GAAW,CqCvCngO,SrCuC6gO,KAAK,GAAW,CqCvC7hO,SrCuCuiO,KAAK,GAAW,CqCvCviO,SrCuCijO,KAAK,GAAW,CqCvCjkO,SrCuC2kO,KAAK,GAAW,CqCvC3lO,SrCuCqmO,KAAK,GAAW,CqCvCrnO,SrCuC+nO,KAAK,GAAW,CqCvC/oO,SrCuCypO,KAAK,GAAW,CqCvCzqO,SrCuCmrO,KAAK,GAAW,CqCvCnsO,SrCuC6sO,KAAK,GAAW,CqCvC7tO,SrCuCuuO,KAAK,GAAW,CqCvCvvO,SrCuCiwO,KAAK,GAAW,CqCvCjxO,SrCuC2xO,KAAK,GAAW,CqCvC3yO,SrCuCqzO,KAAK,GAAW,CqCvCr0O,SrCuC+0O,KAAK,GAAW,CqCvC/1O,SrCuCy2O,KAAK,GAAW,CqCvCz3O,SrCuCm4O,KAAK,GAAW,CqCvCn5O,SrCuC65O,KAAK,GAAW,CqCvC75O,SrCuCu6O,KAAK,GAAW,CqCvCv7O,SrCuCi8O,KAAK,GAAW,CqCvCj9O,SrCuC29O,KAAK,GAAW,CqCvC39O,SrCuCq+O,KAAK,GAAW,CqCvCr/O,SrCuC+/O,KAAK,GAAW,CqCvC//O,SrCuCygP,KAAK,GAAW,CqCvCzhP,SrCuCmiP,KAAK,GAAW,CqCvCniP,SrCuC6iP,KAAK,GAAW,CqCvC7jP,SrCuCukP,KAAK,GAAW,CqCvCvlP,SrCuCimP,KAAK,GAAW,CqCvCjnP,SrCuC2nP,KAAK,GAAW,CqCvC3oP,SrCuCqpP,KAAK,GAAW,CqCvCrqP,SrCuC+qP,KAAK,GAAW,CqCvC/rP,SrCuCysP,KAAK,GAAW,CqCvCztP,SrCuCmuP,KAAK,GAAW,CqCvCnvP,SrCuC6vP,KAAK,GAAW,CqCvC7wP,SrCuCuxP,KAAK,GAAW,CqCvCvyP,SrCuCizP,KAAK,GAAW,CqCvCjzP,SrCuC20P,KAAK,GAAW,CqCvC30P,SrCuCq1P,KAAK,GAAW,CqCvCr2P,SrCuC+2P,KAAK,GAAW,CqCvC/3P,SrCuCy4P,KAAK,GAAW,CqCvCz5P,SrCuCm6P,KAAK,GAAW,CqCvCn7P,SrCuC67P,KAAK,GAAW,CqCvC77P,SrCuCu8P,KAAK,GAAW,CqCvCv9P,SrCuCi+P,KAAK,GAAW,CqCvCj/P,SrCuC2/P,KAAK,GAAW,CqCvC3gQ,SrCuCqhQ,KAAK,GAAW,CqCvCrhQ,SrCuC+hQ,KAAK,GAAW,CqCvC/iQ,SrCuCyjQ,KAAK,GAAW,CqCvCzkQ,SrCuCmlQ,KAAK,GAAW,CqCvCnmQ,SrCuC6mQ,KAAK,GAAW,CqCvC7nQ,SrCuCuoQ,KAAK,GAAW,CqCvCvpQ,SrCuCiqQ,KAAK,GAAW,CqCvCjrQ,SrCuC2rQ,KAAK,GAAW,CqCvC3sQ,SrCuCqtQ,KAAK,GAAW,CqCvCrtQ,SrCuC+tQ,KAAK,GAAW,CqCvC/uQ,SrCuCyvQ,KAAK,GAAW,CqCvCzwQ,SrCuCmxQ,KAAK,GAAW,CqCvCnyQ,SrCuC6yQ,KAAK,GAAW,CqCvC7zQ,SrCuCu0Q,KAAK,GAAW,CqCvCv1Q,SrCuCi2Q,KAAK,GAAW,CqCvCj3Q,SrCuC23Q,KAAK,GAAW,CqCvC33Q,SrCuCq4Q,KAAK,GAAW,CqCvCr5Q,SrCuC+5Q,KAAK,GAAW,CqCvC/6Q,SrCuCy7Q,KAAK,GAAW,CqCvCz8Q,SrCuCm9Q,KAAK,GAAW,CqCvCn+Q,SrCuC6+Q,KAAK,GAAW,CqCvC7/Q,SrCuCugR,KAAK,GAAW,CqCvCvhR,SrCuCiiR,KAAK,GAAW,CqCvCjjR,SrCuC2jR,KAAK,GAAW,CqCvC3kR,SrCuCqlR,KAAK,GAAW,CqCvCrmR,SrCuC+mR,KAAK,GAAW,CqCvC/nR,SrCuCyoR,KAAK,GAAW,CqCvCzpR,SrCuCmqR,KAAK,GAAW,CqCvCnrR,SrCuC6rR,KAAK,GAAW,CqCvC7sR,SrCuCutR,KAAK,GAAW,CqCvCvuR,SrCuCivR,KAAK,GAAW,CqCvCjwR,SrCuC2wR,KAAK,GAAW,CqCvC3xR,SrCuCqyR,KAAK,GAAW,CqCvCryR,SrCuC+yR,KAAK,GAAW,CqCvC/zR,SrCuCy0R,KAAK,GAAW,CqCvCz1R,SrCuCm2R,KAAK,GAAW,CqCvCn3R,SrCuC63R,KAAK,GAAW,CqCvC73R,SrCuCu4R,KAAK,GAAW,CqCvCv5R,SrCuCi6R,KAAK,GAAW,CqCvCj7R,SrCuC27R,KAAK,GAAW,CqCvC37R,SrCuCq8R,KAAK,GAAW,CqCvCr9R,SrCuC+9R,KAAK,GAAW,CqCvC/+R,SrCuCy/R,KAAK,GAAW,CqCvCz/R,SrCuCmgS,KAAK,GAAW,CqCvCngS,SrCuC6gS,KAAK,GAAW,CqCvC7hS,SrCuCuiS,KAAK,GAAW,CqCvCviS,SrCuCijS,KAAK,GAAW,CqCvCjkS,SrCuC2kS,KAAK,GAAW,CqCvC3lS,SrCuCqmS,KAAK,GAAW,CqCvCrnS,SrCuC+nS,KAAK,GAAW,CqCvC/oS,SrCuCypS,KAAK,GAAW,CqCvCzqS,SrCuCmrS,KAAK,GAAW,CqCvCnsS,SrCuC6sS,KAAK,GAAW,CqCvC7tS,SrCuCuuS,KAAK,GAAW,CqCvCvvS,SrCuCiwS,KAAK,GAAW,CqCvCjxS,SrCuC2xS,KAAK,GAAW,CqCvC3yS,SrCuCqzS,KAAK,GAAW,CqCvCr0S,SrCuC+0S,KAAK,GAAW,CqCvC/1S,SrCuCy2S,KAAK,GAAW,CqCvCz3S,SrCuCm4S,KAAK,GAAW,CqCvCn5S,SrCuC65S,KAAK,GAAW,CqCvC75S,SrCuCu6S,KAAK,GAAW,CqCvCv7S,SrCuCi8S,KAAK,GAAW,CqCvCj9S,SrCuC29S,KAAK,GAAW,CqCvC39S,SrCuCq+S,KAAK,GAAW,CqCvCr/S,SrCuC+/S,KAAK,GAAW,CqCvC//S,SrCuCygT,KAAK,GAAW,CqCvCzhT,SrCuCmiT,KAAK,GAAW,CqCvCniT,SrCuC6iT,KAAK,GAAW,CqCvC7jT,SrCuCukT,KAAK,GAAW,CqCvCvlT,SrCuCimT,KAAK,GAAW,CqCvCjnT,SrCuC2nT,KAAK,GAAW,CqCvC3oT,SrCuCqpT,KAAK,GAAW,CqCvCrqT,SrCuC+qT,KAAK,GAAW,CqCvC/rT,SrCuCysT,KAAK,GAAW,CqCvCztT,SrCuCmuT,KAAK,GAAW,CqCvCnvT,SrCuC6vT,KAAK,GAAW,CqCvC7wT,SrCuCuxT,KAAK,GAAW,CqCvCvyT,SrCuCizT,KAAK,GAAW,CqCvCjzT,SrCuC20T,KAAK,GAAW,CqCvC30T,SrCuCq1T,KAAK,GAAW,CqCvCr2T,SrCuC+2T,KAAK,GAAW,CqCvC/3T,SrCuCy4T,KAAK,GAAW,CqCvCz5T,SrCuCm6T,KAAK,GAAW,CqCvCn7T,SrCuC67T,KAAK,GAAW,CqCvC77T,SrCuCu8T,KAAK,GAAW,CqCvCv9T,SrCuCi+T,KAAK,GAAW,CqCvCj/T,SrCuC2/T,KAAK,GAAW,CqCvC3gU,SrCuCqhU,KAAK,GAAW,CqCvCrhU,SrCuC+hU,KAAK,GAAW,CqCvC/iU,SrCuCyjU,KAAK,GAAW,CqCvCzkU,SrCuCmlU,KAAK,GAAW,CqCvCnmU,SrCuC6mU,KAAK,GAAW,CqCvC7nU,SrCuCuoU,KAAK,GAAW,CqCvCvpU,SrCuCiqU,KAAK,GAAW,CqCvCjrU,SrCuC2rU,KAAK,GAAW,CqCvC3sU,SrCuCqtU,KAAK,GAAW,CqCvCrtU,SrCuC+tU,KAAK,GAAW,CqCvC/uU,SrCuCyvU,KAAK,GAAW,CqCvCzwU,SrCuCmxU,KAAK,GAAW,CqCvCnyU,SrCuC6yU,KAAK,GAAW,CqCvC7zU,SrCuCu0U,KAAK,GAAW,CqCvCv1U,SrCuCi2U,KAAK,GAAW,CqCvCj3U,SrCuC23U,KAAK,GAAW,CqCvC33U,SrCuCq4U,KAAK,GAAW,CqCvCr5U,SrCuC+5U,KAAK,GAAW,CqCvC/6U,SrCuCy7U,KAAK,GAAW,CqCvCz8U,SrCuCm9U,KAAK,GAAW,CqCvCn+U,SrCuC6+U,KAAK,GAAW,CqCvC7/U,SrCuCugV,KAAK,GAAW,CqCvCvhV,SrCuCiiV,KAAK,GAAW,CqCvCjjV,SrCuC2jV,KAAK,GAAW,CqCvC3kV,SrCuCqlV,KAAK,GAAW,CqCvCrmV,SrCuC+mV,KAAK,GAAW,CqCvC/nV,SrCuCyoV,KAAK,GAAW,CqCvCzpV,SrCuCmqV,KAAK,GAAW,CqCvCnrV,SrCuC6rV,KAAK,GAAW,CqCvC7sV,SrCuCutV,KAAK,GAAW,CqCvCvuV,SrCuCivV,KAAK,GAAW,CqCvCjwV,SrCuC2wV,KAAK,GAAW,CqCvC3xV,SrCuCqyV,KAAK,GAAW,CqCvCryV,SrCuC+yV,KAAK,GAAW,CqCvC/zV,SrCuCy0V,KAAK,GAAW,CqCvCz1V,SrCuCm2V,KAAK,GAAW,CqCvCn3V,SrCuC63V,KAAK,GAAW,CqCvC73V,SrCuCu4V,KAAK,GAAW,CqCvCv5V,SrCuCi6V,KAAK,GAAW,CqCvCj7V,SrCuC27V,KAAK,GAAW,CqCvC37V,SrCuCq8V,KAAK,GAAW,CqCvCr9V,SrCuC+9V,KAAK,GAAW,CqCvC/+V,SrCuCy/V,KAAK,GAAW,CqCvCz/V,SrCuCmgW,KAAK,GAAW,CqCvCngW,SrCuC6gW,KAAK,GAAW,CqCvC7hW,SrCuCuiW,KAAK,GAAW,CqCvCviW,SrCuCijW,KAAK,GAAW,CqCvCjkW,SrCuC2kW,KAAK,GAAW,CqCvC3lW,SrCuCqmW,KAAK,GAAW,CqCvCrnW,SrCuC+nW,KAAK,GAAW,CqCvC/oW,SrCuCypW,KAAK,GAAW,CqCvCzqW,SrCuCmrW,KAAK,GAAW,CqCvCnsW,SrCuC6sW,KAAK,GAAW,CqCvC7tW,SrCuCuuW,KAAK,GAAW,CqCvCvvW,SrCuCiwW,KAAK,GAAW,CqCvCjxW,SrCuC2xW,KAAK,GAAW,CqCvC3yW,SrCuCqzW,KAAK,GAAW,CqCvCr0W,SrCuC+0W,KAAK,GAAW,CqCvC/1W,SrCuCy2W,KAAK,GAAW,CqCvCz3W,SrCuCm4W,KAAK,GAAW,CqCvCn5W,SrCuC65W,KAAK,GAAW,CqCvC75W,SrCuCu6W,KAAK,GAAW,CqCvCv7W,SrCuCi8W,KAAK,GAAW,CqCvCj9W,SrCuC29W,KAAK,GAAW,CqCvC39W,SrCuCq+W,KAAK,GAAW,CqCvCr/W,SrCuC+/W,KAAK,GAAW,CqCvC//W,SrCuCygX,KAAK,GAAW,CqCvCzhX,SrCuCmiX,KAAK,GAAW,CqCvCniX,SrCuC6iX,KAAK,GAAW,CqCvC7jX,SrCuCukX,KAAK,GAAW,CqCvCvlX,SrCuCimX,KAAK,GAAW,CqCvCjnX,SrCuC2nX,KAAK,GAAW,CqCvC3oX,SrCuCqpX,KAAK,GAAW,CqCvCrqX,SrCuC+qX,KAAK,GAAW,CqCvC/rX,SrCuCysX,KAAK,GAAW,CqCvCztX,SrCuCmuX,KAAK,GAAW,CqCvCnvX,SrCuC6vX,KAAK,GAAW,CqCvC7wX,SrCuCuxX,KAAK,GAAW,CqCvCvyX,SrCuCizX,KAAK,GAAW,CqCvCjzX,SrCuC20X,KAAK,GAAW,CqCvC30X,SrCuCq1X,KAAK,GAAW,CqCvCr2X,SrCuC+2X,KAAK,GAAW,CqCvC/3X,SrCuCy4X,KAAK,GAAW,CqCvCz5X,SrCuCm6X,KAAK,GAAW,CqCvCn7X,SrCuC67X,KAAK,GAAW,CqCvC77X,SrCuCu8X,KAAK,GAAW,CqCvCv9X,SrCuCi+X,KAAK,GAAW,CqCvCj/X,SrCuC2/X,KAAK,GAAW,CqCvC3gY,SrCuCqhY,KAAK,GAAW,CqCvCrhY,SrCuC+hY,KAAK,GAAW,CqCvC/iY,SrCuCyjY,KAAK,GAAW,CqCvCzkY,SrCuCmlY,KAAK,GAAW,CqCvCnmY,SrCuC6mY,KAAK,GAAW,CqCvC7nY,SrCuCuoY,KAAK,GAAW,CqCvCvpY,SrCuCiqY,KAAK,GAAW,CqCvCjrY,SrCuC2rY,KAAK,GAAW,CqCvC3sY,SrCuCqtY,KAAK,GAAW,CqCvCrtY,SrCuC+tY,KAAK,GAAW,CqCvC/uY,SrCuCyvY,KAAK,GAAW,CqCvCzwY,SrCuCmxY,KAAK,GAAW,CqCvCnyY,SrCuC6yY,KAAK,GAAW,CqCvC7zY,SrCuCu0Y,KAAK,GAAW,CqCvCv1Y,SrCuCi2Y,KAAK,GAAW,CqCvCj3Y,SrCuC23Y,KAAK,GAAW,CqCvC33Y,SrCuCq4Y,KAAK,GAAW,CqCvCr5Y,SrCuC+5Y

BAAJ,C;QACI,OAAO,I;MACX,OAAc,YAAP,MAAO,EAAU,SAAV,C;K;oFAGlB,8B;MASI,OAAO,WAAW,IA
AX,IAAmB,2BAAS,OAAT,C;K;oFAG9B,8B;MASI,OAAO,WAAW,IAAX,IAAmB,2BAAS,OAAT,C;K;IAG9B,u
C;MAMI,OAAO,2BvC4K4B,SuC5KnB,KvC4K6B,KAAL,GAAiB,GAAtB,CuC5K5B,C;K;IAGX,uC;MAMI,OAA
O,2BvC6K8B,UAAW,oBuC7KhC,KvC6K2B,KAAK,CAAL,UAAN,CuC7K9B,C;K;IAGX,uC;MAMI,OAAO,2Bt
CwL8B,UAAW,oBsCxLhC,KtCwL2B,KAAK,CAAL,iBAAN,CsCxL9B,C;K;IAGX,uC;MAMY,Q;MAAD,cAAC,
OtBqF4C,UsBrF5C,KtBqFkD,yBsBrFxC,EtBqFwC,CAAN,CsBrF7C,wBAA8B,2BAA9B,Q;MAAA,W;QAAqC,o
CtCoPR,SsCpPiB,KtB6KlB,KhBuEW,QAAV,CsCpPQ,C;;MAA5C,a;K;IAGJ,uC;MAMI,OAAO,2BrCyI4B,SqCzI
nB,KrCyI6B,KAAL,GAAiB,KAAtB,CqCzI5B,C;K;IAGX,uC;MAMI,OAAO,2BrC0I8B,UAAW,oBqC1IhC,KrC0I
2B,KAAK,CAAL,YAAN,CqC1I9B,C;K;IAGX,kC;MASI,OAAO,uCAAgB,yBvCmHY,SuCnHI,SvCmHM,KAAL,
GAAiB,GAAtB,CuCnHZ,EvCmHY,SuCnHmB,EvCmHT,KAAL,GAAiB,GAAtB,CuCnHZ,EAA4C,EAA5C,C;K;I
AG3B,kC;MASI,OAAO,uCAAgB,yBAAgB,SAAhB,EAAsB,EAAtB,EAA0B,EAA1B,C;K;IAG3B,kC;MASI,OAA
O,wCAAiB,yBAAgB,SAAhB,EAAsB,EAAtB,M;K;IAG5B,kC;MASI,OAAO,uCAAgB,yBrCgFY,SqChFI,SrCgF
M,KAAL,GAAiB,KAAtB,CqChFZ,ErCgFY,SqChFmB,ErCgFT,KAAL,GAAiB,KAAtB,CqChFZ,EAA4C,EAA5C,
C;K;IAG3B,gC;MAMI,OAAO,uCAAgB,yBAAgB,cAAhB,EAAwB,eAAtB,EAA6B,CAAC,cAAD,IAA7B,C;K;IA
G3B,gC;MAMI,OAAO,wCAAiB,yBAAgB,cAAhB,EAAwB,eAAtB,EAA8B,cAAD,aAA7B,C;K;IAG5B,iC;MAMI,
oBAAoB,OAAO,CAA3B,EAA8B,IAA9B,C;MACA,OAAO,uCAAgB,yBAAgB,eAAhB,EAAwB,EAAuB,cAAvB,EAAiC,S
AAK,KAAL,GAAY,CAAhB,EAAmB,GAAnB,IAAwB,GAA6B,CAAC,IAAD,IAAO,C;K;IAG3B,iC;MAMI,oBAAoB,kB
AAO,CAA3B,EAA8B,IAA9B,C;MACA,OAAO,wCAAiB,yBAAgB,eAAhB,EAAwB,EAAuB,cAAvB,EAAiC,SAAK,KAA
L,cAAY,CAAhB,EAAmB,GAAnB,IAAwB,GAA8B,IAAD,aAA1D,C;K;IAG5B,iC;MAQI,IvC/OgF,0BuC+O5E,EvCJkC,K
AAL,GAAiB,GA3O8B,EuC+O5E,OtE,6BAAM,UvCJsB,KAAL,GAAiB,GA3O8B,CuC+O5E,KAAJ,C;QAA2B,OAAO,
iCAAU,M;MAChC,WvC6BuB,SuC7B5B,SvC6BsC,KAAL,GAAiB,GAAtB,CuC7BjC,C;MuC7BV,YAAK,W;MAA9B,OtCj
D6D,oBAhJP,SAAU,CD8N7B,SuC7BV,EvC6BoB,KAAL,GAAiB,GAAtB,CC9N6B,MAAK,GDAK,KCAO,KAA
Z,IAAf,CAgJO,C;K;IsCoDjE,iC;MAQI,ItC3OkE,YsC2O9D,EtC3O+E,KAAjB,EsC2OxD,4BAAK,UtC3OgF,KAA
7B,CsC2O9D,KAAJ,C;QAA0B,OAAO,iCAAU,M;MAC3C,OtC7D6D,csC6DtD,StC7DsD,EAhJP,SsC6MtC,EtC7
MgD,KAAK,GAAY,CsC6M5D,WtC7M4D,MAAZ,IAAf,CAgJO,C;K;IsCgEjE,iC;MAQI,ItB/OmE,asB+O/D,EtB/
OiF,KAAlB,EsB+OzD,6BAAM,UtB/OiF,KAA9B,CsB+O/D,KAAJ,C;QAA2B,OAAO,kCAAW,M;MAC7C,OtBzE
+D,iBsByExD,StBzEwD,EA7IP,UsBsNxC,EtBtNmD,KAAK,UAAY,ChByP/C,UAAW,oBAAL,CsCnCtB,WtCmC
sB,MAAK,GAAL,CAAiB,IAAN,CgBnP+C,MAAZ,CAAhB,CA6IO,CA6IP,CA6IO,C;K;IsB4EnE,iC;MAQI,IrC3QiF,0BqC2Q7E,ErCvCvC
kC,KAAL,GAAiB,GAAtB,CqCuCZ,EAAO,KApO+B,EqC2Q7E,8BAAO,UrCvCqB,KAAL,GAAiB,GAAtB,CqCuCZ,EAAO,KApO+B,CqC2Q7E,KAAJ,C;QAA4B,OAAO,
OAAO,iCAAU,M;MACjC,WrCNuB,SqCM5B,SrCNsC,KAAL,GAAiB,GAAtB,CqCMjC,C;MqCMV,YAAK,W;MAA9B,OtCr
F6D,oBAhJP,SAAU,CC+N7B,SqCMV,ErCNoB,KAAL,GAAiB,GAAtB,CC/N6B,MAAK,GDAK,KCAO,KAA
Z,IAAf,CAgJO,CAgJP,CAgJO,C;K;IsCoEjE,iC;MAQI,IvC/OmE,asB+O/D,EvC/
OiF,KAAlB,EsB+OzD,6BAAM,UvC/OiF,KA9B,CsB+O/D,KAAJ,C;QAA2B,OAAO,kCAAW,M;MAC7C,OvCzE
+D,iBsByExD,StBzEwD,EA7IP,UsBsNxC,EvCtNmD,KAAK,UAAY,ChByP/C,UAAW,oBAAL,CsCnCtB,WvCmC
sB,MAAK,GAAL,CAAiB,IAAN,CgBnP+C,MAAZ,CAAhB,CA6IO,CA6IP,CA6IO,C;K;IsB4EnE,iC;MAQI,IrC3QiF,0BqC2Q7E,ErCvCvC
kC,KAAL,GAAiB,GAAtB,CqCuCZ,EAAO,KApO+B,EqC2Q7E,8BAAO,UrCvCqB,KAAL,GAAiB,GAAtB,CqCuCZ,EAAO,KApO+B,CqC2Q7E,KAAJ,C;QAA4B,OAAO,
OAAO,iCAAU,M;MACjC,WrCNuB,SqCM5B,SrCNsC,KAAL,GAiB,GAAtB,CqCMjC,C;MqCMV,YAAK,W;MAA9B,OtCr
F6D,oBAhJP,SAAU,CC+N7B,SqCMV,ErCNoB,KAAL,GAiB,GAAtB,CC/N6B,MAAK,GDAK,KCAO,KAA
Z,IAAf,CAgJO,CAgJP,CAgJO,C;K;IsCoEjE,iC;MAQI,IvC/OmE,asB+O/D,EvC/
OiF,KAAlB,EsB+OzD,6BAAM,UvC/OiF,KA9B,CsB+O/D,KAAJ,C;QAA2B,OAAO,kCAAW,M;MAC7C,OvCzE
+D,iBsByExD,StBzEwD,EA7IP,UsBsNxC,EvCtNmD,KAAK,UAAY,ChByP/C,UAAW,oBAAL,CsCnCtB,WvCmC
sB,MAAK,GAAL,CAAiB,IAAN,CgBnP+C,MAAZ,CAAhB,CA6IO,CA6IP,CA6IO,C;K;IsB4EnE,iC;MAQI,IrC3QiF,0BqC2Q7E,ErCvCvC
kC,KAAL,GAiB,GAAtB,CqCuCZ,EAAO,KApO+B,EqC2Q7E,8BAAO,UrCvCqB,KAAL,G

B,OAAO,Y;MAChC,ItBnYmE,asBmY/D,StBnYiF,KAAlB,EsBmYxD,YtBnYsF,KAA9B,CsBmY/D,IAAJ,C;QAA
yB,OAAO,Y;MAChC,OAAO,S;K;IAGX,4D;MAUI,IvCzagF,0BuCya5E,YvC9LkC,KAAL,GAAiB,GA3O8B,EuC
ya7D,YvC9LmB,KAAL,GAAiB,GA3O8B,CuCya5E,IAAJ,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAo
F,YAApF,MAAzB,C;MACvC,IvC1agF,0BuC0a5E,SvC/LkC,KAAL,GAAiB,GA3O8B,EuC0arE,YvC/L2B,KAAL,
GAAiB,GA3O8B,CuC0a5E,IAAJ,C;QAAyB,OAAO,Y;MAChC,IvC3agF,0BuC2a5E,SvChMkC,KAAL,GAAiB,G
A3O8B,EuC2arE,YvChM2B,KAAL,GAAiB,GA3O8B,CuC2a5E,IAAJ,C;QAAyB,OAAO,Y;MAChC,OAAO,S;K;I
AGX,4D;MAUI,IrCjbiF,0BqCib7E,YrC7MkC,KAAL,GAAiB,KApO+B,EqCib9D,YrC7MmB,KAAL,GAAiB,KAp
O+B,CqCib7E,IAAJ,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IrClbiF,
0BqCkb7E,SrC9MkC,KAAL,GAAiB,KApO+B,EqCkbtE,YrC9M2B,KAAL,GAAiB,KApO+B,CqCkb7E,IAAJ,C;
QAAyB,OAAO,Y;MAChC,IrCnbiF,0BqCmb7E,SrC/MkC,KAAL,GAAiB,KApO+B,EqCmbtE,YrC/M2B,KAAL,G
AAiB,KApO+B,CqCmb7E,IAAJ,C;QAAyB,OAAO,Y;MAChC,OAAO,S;K;IAGX,uC;MAcW,Q;MAJP,IAAI,8CA
AJ,C;QACI,OAAY,WAAL,SAAK,EAAe,KAAf,C;;MAEhB,IAAI,KAAM,UAAV,C;QAAqB,MAAM,gCAAyB,4C
AAyC,KAAzC,MAAzB,C;MAEvB,ItC9b8D,YsC8b9D,StC9b+E,KAAjB,EsC8bvD,KAAM,MtC9b8E,KAA7B,Cs
C8b9D,K;QAA4B,OAAN,KAAM,M;;QAC5B,ItC/b8D,YsC+b9D,StC/b+E,KAAjB,EsC+bvD,KAAM,atC/b8E,KA
A7B,CsC+b9D,K;UAAmC,OAAN,KAAM,a;;UAC3B,gB;;MAHZ,W;K;IAOJ,uC;MAcW,Q;MAJP,IAAI,8CAAJ,C
;QACI,OAAY,WAAL,SAAK,EAAgB,KAAhB,C;;MAEhB,IAAI,KAAM,UAAV,C;QAAqB,MAAM,gCAAyB,4C
AAyC,KAAzC,KAAzB,C;MAEvB,ItB3c+D,asB2c/D,StB3ciF,KAAlB,EsB2cxD,KAAM,MtB3cgF,KAA9B,CsB2c
/D,K;QAA4B,OAAN,KAAM,M;;QAC5B,ItB5c+D,asB4c/D,StB5ciF,KAAlB,EsB4cxD,KAAM,atB5cgF,KAA9B,
CsB4c/D,K;UAAmC,OAAN,KAAM,a;;UAC3B,gB;;MAHZ,W;K;IC/fJ,2B;MAUoB,Q;MADhB,UAAgB,W;MAC
A,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MvCoDiD,SuCpDjD,GvCoD2D,KAAK,GuCpDzD,OvCoDo
E,KAAX,IAAf,C;;MuClDrD,OAAO,G;K;IAGX,2B;MAUoB,Q;MADhB,UAAiB,2B;MACD,2B;MAAhB,OAAgB,
cAAhB,C;QAAgB,yB;QACZ,MvBuCmD,SuBvCnD,GvBuC8D,KAAK,KuBvC5D,OvBuCuE,KAAX,CAAhB,C;;
MuBrCvD,OAAO,G;K;IAGX,2B;MAUoB,Q;MADhB,UAAgB,W;MACA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,
yB;QACZ,MvCoBiD,SuCpBjD,GvCoB2D,KAAK,GAAW,CD2O5C,SwC/PxB,OxC+PkC,KAAL,GAAiB,GAAtB,
CC3O4C,MAAX,IAAf,C;;MuClBrD,OAAO,G;K;IAGX,2B;MAUoB,Q;MADhB,UAAgB,W;MACA,2B;MAAhB,
OAAgB,cAAhB,C;QAAgB,yB;QACZ,MvCIiD,SuCJjD,GvCI2D,KAAK,GAAW,CC4O5C,SsChPxB,OtCgPkC,KA
AL,GAAiB,KAAtB,CD5O4C,MAAX,IAAf,C;;MuCFrD,OAAO,G;K;;;;;ICuCP,iD;MAAA,qE;MAAgB,4B;MANp
B,uC;MAMI,Y;K;IACA,4D;MAAA,qE;MAAgC,wBAAM,OAAN,Q;MAPpC,uC;MAOI,Y;K;IACA,mE;MAAA,q
E;MAAmD,6BAAM,OAAN,EAAe,KAAf,C;MARvD,uC;MAQI,Y;K;IACA,0D;MAAA,qE;MAAiC,wBAAM,KA
AN,Q;MATrC,uC;MASI,Y;K;ICxGJ,gC;K;;;;ICuBoC,wC;8BAAsC,O;;K;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;yCC0RtE,6B;MA
SI,MAAM,yB;K;;;;;;;0CAyDV,sB;MASI,OAAO,I;K;;;;;;;;;;;;;;;;;;ICnYf,wB;K;kCAEI,Y;MAA4B,sB;K;;IAMhC,
wB;K;kCAEI,Y;MAA4B,mC;K;;IAMhC,yB;K;mCAEI,Y;MAA4B,uB;K;;IAMhC,uB;K;iCAEI,Y;MAA4B,qB;K;;I
AMhC,wB;K;kCAEI,Y;MAA4B,sB;K;;IAMhC,yB;K;mCAEI,Y;MAA4B,uB;K;;IAMhC,0B;K;oCAEI,Y;MAA4B,
wB;K;;IAMhC,2B;K;qCAEI,Y;MAA4B,yB;K;;ICtDM,oD;MAA2C,uB;MAAjB,gB;MAC5D,sBAAgC,InBkCU,I;
MmBjC1C,iBAAmC,YAAO,CAAX,GAAc,SAAS,IAAvB,gB,GAAiC,SAAS,I;MACzE,cAAuB,KnB
gCK,ImBhC1C,GAAqD,mB;K;gDAAE,rD,Y;MAAkC,qB;K;iDAAlC,Y;MACI,YAAY,W;MACZ,IAAI,UAAS,mBA
Ab,C;QACI,IAAI,CAAC,cAAL,C;UAAc,MAAa,6B;QAC3B,iBAAU,K;;QAGV,4BAAW,I;;MAEf,OAAO,KAAM,C;K;;IAQb,gB
gB,mD;MAAyC,sB;MAAjB,gB;MACpC,sBAAgC,I;MAAhC,cAAuB,KAAM,C;MAAb,OAAO,I;K;;IAGvB,iBAAmC,YAAO,CAAX,GAAc,SAAS,IAAvB,gB,GAAiC,SAAS,I;MACzE,cAAuB,KAAM,C;MAAb,gBAAO,I;MAAP

Q,SAAtB,GAAgC,aAAQ,S;K;uCAE7E,iB;MACI,iDAA6B,kBAAa,KAAM,UAAnB,KAC7B,eAAS,KAAM,MAAf
,IAAwB,cAAQ,KAAM,KAAtC,IAA8C,cAAQ,KAAM,KAD/B,CAA7B,C;K;yCAGJ,Y;MACI,OAAI,cAAJ,GAAe,
EAAf,GAAwB,OAAM,OAAK,UpBRG,IoBQR,UAAkB,SpBRV,IoBQR,KAAN,SAAqC,SAArC,I;K;yCAE5B,Y;
MAAkC,OAAI,YAAO,CAAX,GAAc,oBAAE,UAAF,+BAAU,SAAV,eAAqB,SAAnC,GAA8C,oBAAE,UAAF,qC
AAgB,SAAhB,gBAA4B,CAAC,SAAD,IAA5B,C;K;IAEhF,qC;MAAA,yC;K;kEACI,sC;MAQ2F,2BAAgB,UAAh
B,EA4B,QA5B,EAAsC,IAAtC,C;K;;;IAT/F,iD;MAAA,gD;QAAA,+B;;MAAA,yC;K;;IAiBA,mD;MA6CA,sC;
MAtCI,IAAI,SAAQ,CAAZ,C;QAAe,MAAa,gCAAyB,wBAAzB,C;MAC5B,IAAI,SAAQ,WAAZ,C;QA2B,MAA
a,gCAAyB,wEAAzB,C;MAG5C,aAGwB,K;MAExB,YAGuB,0BAA0B,KA1B,EAAiC,YAAjC,EAA+C,IAA/C,C
;MAEvB,YAGuB,I;K;wCAEvB,Y;MAAuC,kCAAuB,UAAvB,EAA8B,SA9B,EAAoC,SAApC,C;K;uCAEvC,Y;
MAMqC,OAAI,YAAO,CAAX,GAAc,aAAQ,SAAtB,GAAgC,aAAQ,S;K;sCAE7E,iB;MACI,gDAA4B,kBAAa,KA
AM,UAAnB,KAC5B,eAAS,KAAM,MAAf,IAAwB,cAAQ,KAAM,KAAtC,IAA8C,cAAQ,KAAM,KADhC,CAA5B,C;K;wCAGJ,Y;MACI,OAAI,cAAJ,GAAe,EAAf,GAAwB,OAAM,MAAK,UAAL,QAAa,SAAb,IAAsB,SAAtB,GAAiC,SAAjC,I
AA4B,SAA5B,GAAuC,SAAvC,GAAkD,SAAlD,GAA6D,SAA7D,GAAwE,SAAxE,GAAmF,SAAnF,I;K;wCAE5B,Y;MAAuC,kCAAuB,UAAvB,EAA8B,SAAS,SAAvC,EAAkD,SAAlD,GAA6D,SAA7D,GAAwE,SAAxE,GAAmF,SAAnF,I;K;;IAAiB,oD;MA6CA,sC;
MAtCI,IAAI,gBAAQ,CAAZ,C;QAAe,MAAa,gCAAyB,wBAAzB,C;MAC5B,IAAI,sCAAQ,CAAZ,C;QA2B,MAA
a,gCAAyB,wEAAzB,C;MAG5C,aAGwB,K;MAExB,YAGuB,0BAA0B,KA1B,EAAiC,YAAjC,EAA+C,IAA/C,C
;MAEvB,YAGuB,I;K;yCAEvB,Y;MAAuC,mCAAuB,UAAvB,EAA8B,SA9B,EAAoC,SAApC,C;K;wCAEvC,Y;MAMqC
,OAAI,uBAAQ,CAAZ,GAAc,2BAAQ,SAAR,GAAkB,2BAAQ,SAAR,GAAkB,uCAAE,SAAF,GAAa,SAAb,GAAwB,SAAxB,GAAmC,SAAnC,GAA8C,SAA9C,GAAyD,SAAzD,GAAoE,SAApE,GAA+E,SAA/E,GAA0F,SAA1F,GAAqG,SAArG,GAAgH,SAAhH,GAA2H,SAA3H,GAAsI,SAAtI,GAAiJ,SAAjJ,GAA4J,SAA5J,GAAuK,SAAvK,GAAkL,SAAlL,GAA6L,SAA7L,GAAwM,SAAxM,GAAmN,SAAnN,GAA8N,SAA9N,GAAyO,SAAzO,GAAoP,SAApP,GAA+P,SAA/P,GAA0Q,SAA1Q,GAAqR,SAArR,GAAgS,SAAhS,GAA2S,SAA3S,GAAsT,SAAtT,GAAiU,SAAjU,GAA4U,SAA5U,GAAuV,SAAvV,GAAkW,SAAlW,GAA6W,SAA7W,GAAwX,SAAxX,GAAmY,SAAnY,GAA8Y,SAA9Y,GAAyZ,SAAzZ,GAAoa,SAApa,GAA+a,SAA/a,GAA0b,SAA1b,GAAqc,SAArc,GAAgd,SAAhd,GAA2d,SAA3d,GAAse,SAAte,GAAif,SAAjf,GAA4f,SAA5f,GAAug,SAAvg,GAAkh,SAAlh,GAA6h,SAA7h,GAAwi,SAAxi,GAAmj,SAAnj,GAA8j,SAA9j,GAAyk,SAAzk,GAAol,SAApl,GAA+l,SAA/l,GAA0m,SAA1m,GAAqn,SAArn,GAAgo,SAAho,GAA2o,SAA3o,GAAsp,SAAtp,GAAiq,SAAjq,GAA4q,SAA5q,GAAur,SAAvr,GAAks,SAAls,GAA6s,SAA7s,GAAwt,SAAxt,GAAmu,SAAnu,GAA8u,SAA9u,GAAyv,SAAzv,GAAow,SAApw,GAA+w,SAA/w,GAA0x,SAA1x,GAAqy,SAAry,GAAgz,SAAhz,GAA2z,SAA3z,GAAs0,SAAt0,GAAi1,SAAj1,GAA41,SAA51,GAAu2,SAAv2,GAAk3,SAAl3,GAA63,SAA73,GAAw4,SAAx4,GAAm5,SAAn5,GAA85,SAA95,GAAy6,SAAz6,GAAo7,SAAp7,GAA+7,SAA/7,GAA08,SAA18,GAAq9,SAAr9,GAAg+,SAAh+,GAA2+,SAA3+,GAAs/,SAAt/,GAAigB,SAAjgB,GAA4gB,SAA5gB,GAAuhB,SAAvhB,GAAkiB,SAAliB,GAA6iB,SAA7iB,GAAwjB,SAAxjB,GAAmkB,SAAnkB,GAA8kB,SAA9kB,GAAymB,SAAzmB,GAAonB,SAApnB,GAA+nB,SAA/nB,GAA0oB,SAA1oB,GAAqpB,SAArpB,GAAgqB,SAAhqB,GAA2qB,SAA3qB,GAAsrB,SAAtrB,GAAisB,SAAjsB,GAA4sB,SAA5sB,GAAutB,SAAvtB,GAAkuB,SAAluB,GAA6uB,SAA7uB,GAAwvB,SAAxvB,GAAmwB,SAAnwB,GAA8wB,SAA9wB,GAAyxB,SAAzxB,GAAoyB,SAApyB,GAA+yB,SAA/yB,GAA0zB,SAA1zB,GAAq0B,SAAr0B,GAAg1B,SAAh1B,GAA21B,SAA31B,GAAs2B,SAAt2B,GAAi3B,SAAj3B,GAA43B,SAA53B,GAAu4B,SAAv4B,GAAk5B,SAAl5B,GAA65B,SAA75B,GAAw6B,SAAx6B,GAAm7B,SAAn7B,GAA87B,SAA97B,GAAy8B,SAAz8B,GAAo9B,SAAp9B,GAA+9B,SAA/9B,GAA0+B,SAA1+B,GAAq/B,SAAr/B,GAAggC,SAAhgC,GAA2gC,SAA3gC,GAAshC,SAAthC,GAAiiC,SAAjiC,GAA4iC,SAA5iC,GAAujC,SAAvjC,GAAkkC,SAAlkC,GAA6kC,SAA7kC,GAAwlC,SAAxlC,GAAmmC,SAAnmC,GAA8mC,SAA9mC,GAAynC,SAAznC,GAAooC,SAApoC,GAA+oC,SAA/oC,GAA0pC,SAA1pC,GAAqqC,SAArqC,GAAgrC,SAAhrC,GAA2rC,SAA3rC,GAAssC,SAAtsC,GAAitC,SAAjtC,GAA4tC,SAA5tC,GAAuuC,SAAvuC,GAAkvC,SAAlvC,GAA6vC,SAA7vC,GAAwwC,SAAxwC,GAAmxC,SAAnxC,GAA8xC,SAA9xC,GAAyyC,SAAzyC,GAAozC,SAApzC,GAA+zC,SAA/zC,GAA00C,SAA10C,GAAq1C,SAAr1C,GAAg2C,SAAh2C,GAA22C,SAA32C,GAAs3C,SAAt3C,GAAi4C,SAAj4C,GAA44C,SAA54C,GAAu5C,SAAv5C,GAAk6C,SAAl6C,GAA66C,SAA76C,GAAw7C,SAAx7C,GAAm8C,SAAn8C,GAA88C,SAA98C,GAAy9C,SAAz9C,GAAo+C,SAAp+C,GAA++C,SAA/+C,GAA0/C,SAA1/C,GAAqgD,SAArgD,GAAghD,SAAhhD,GAA2hD,SAA3hD,GAAsiD,SAAtiD,GAAijD,SAAjjD,GAA4jD,SAA5jD,GAAukD,SAAvkD,GAAklD,SAAllD,GAA6lD,SAA7lD,GAAwmD,SAAxmD,GAAmnD,SAAnnD,GAA8nD,SAA9nD,GAAyoD,SAAzoD,GAAopD,SAAppD,GAA+pD,SAA/pD,GAA0qD,SAA1qD,GAAqrD,SAArrD,GAAgsD,SAAhsD,GAA2sD,SAA3sD,GAAstD,SAAttD,GAAiuD,SAAjuD,GAA4uD,SAA5uD,GAAuvD,SAAvvD,GAAkwD,SAAlwD,GAA6wD,SAA7wD,GAAwxD,SAAxxD,GAAmyD,SAAnyD,GAA8yD,SAA9yD,GAAyzD,SAAzzD,GAAo0D,SAAp0D,GAA+0D,SAA/0D,GAA01D,SAA11D,GAAq2D,SAAr2D,GAAg3D,SAAh3D,GAA23D,SAA33D,GAAs4D,SAAt4D,GAAi5D,SAAj5D,GAA45D,SAA55D,GAAu6D,SAAv6D,GAAk7D,SAAl7D,GAA67D,SAA77D,GAAw8D,SAAx8D,GAAm9D,SAAn9D,GAA89D,SAA99D,GAAy+D,SAAz+D,GAAo/D,SAAp/D,GAA+/D,SAA//D,GAA0gE,SAA1gE,GAAqhE,SAArhE,GAAgiE,SAAhiE,GAA2iE,SAA3iE,GAAsjE,SAAtjE,GAAikE,SAAjkE,GAA4kE,SAA5kE,GAAulE,SAAvlE,GAAkmE,SAAlmE,GAA6mE,SAA7mE,GAAwnE,SAAxnE,GAAmoE,SAAnoE,GAA8oE,SAA9oE,GAAypE,SAAzpE,GAAoqE,SAApqE,GAA+qE,SAA/qE,GAA0rE,SAA1rE,GAAqsE,SAArsE,GAAgtE,SAAhtE,GAA2tE,SAA3tE,GAAsuE,SAAtuE,GAAivE,SAAjvE,GAA4vE,SAA5vE,GAAuwE,SAAvwE,GAAkxE,SAAlxE,GAA6xE,SAA7xE,GAAwyE,SAAxyE,GAAmzE,SAAnzE,GAA8zE,SAA9zE,GAAy0E,SAAz0E,GAAo1E,SAAp1E,GAA+1E,SAA/1E,GAA02E,SAA12E,GAAq3E,SAAr3E,GAAg4E,SAAh4E,GAA24E,SAA34E,GAAs5E,SAAt5E,GAAi6E,SAAj6E,GAA46E,SAA56E,GAAu7E,SAAv7E,GAAk8E,SAAl8E,GAA68E,SAA78E,GAAw9E,SAAx9E,GAAm+E,SAAn+E,GAA8+E,SAA9+E,GAAy/E,SAAz/E,GAAogF,SAApgF,GAA+gF,SAA/gF,GAA0hF,SAA1hF,GAAqiF,SAAriF,GAAgjF,SAAhjF,GAA2jF,SAA3jF,GAAskF,SAAtkF,GAAilF,SAAjlF,GAA4lF,SAA5lF,GAAumF,SAAvmF,GAAknF,SAAlnF,GAA6nF,SAA7nF,GAAwoF,SAAxoF,GAAmpF,SAAnpF,GAA8pF,SAA9pF,GAAyqF,SAAzqF,GAAorF,SAAprF,GAA+rF,SAA/rF,GAA0sF,SAA1sF,GAAqtF,SAArtF,GAAguF,SAAhuF,GAA2uF,SAA3uF,GAAsvF,SAAtvF,GAAiwF,SAAjwF,GAA4wF,SAA5wF,GAAuxF,SAAvxF,GAAkyF,SAAlyF,GAA6yF,SAA7yF,GAAwzF,SAAxzF,GAAm0F,SAAn0F,GAA80F,SAA90F,GAAy1F,SAAz1F,GAAo2F,SAAp2F,GAA+2F,SAA/2F,GAA03F,SAA13F,GAAq4F,SAAr4F,GAAg5F,SAAh5F,GAA25F,SAA35F,GAAs6F,SAAt6F,GAAi7F,SAAj7F,GAA47F,SAA57F,GAAu8F,SAAv8F,GAAk9F,SAAl9F,GAA69F,SAA79F,GAAw+F,SAAx+F,GAAm/F,SAAn/F,GAA8/F,SAA9/F,GAAygG,SAAzgG,GAAohG,SAAphG,GAA+hG,SAA/hG,GAA0iG,SAA1iG,GAAqjG,SAArjG,GAAgkG,SAAhkG,GAA2kG,SAA3kG,GAAslG,SAAtlG,GAAimG,SAAjmG,GAA4mG,SAA5mG,GAAunG,SAAvnG,GAAkoG,SAAloG,GAA6oG,SAA7oG,GAAwpG,SAAxpG,GAAmqG,SAAnqG,GAA8qG,SAA9qG,GAAyrG,SAAzrG,GAAosG,SAApsG,GAA+sG,SAA/sG,GAA0tG,SAA1tG,GAAquG,SAAruG,GAAgvG,SAAhvG,GAA2vG,SAA3vG,GAAswG,SAAtwG,GAAixG,SAAjxG,GAA4xG,SAA5xG,GAAuyG,SAAvyG,GAAkzG,SAAlzG,GAA6zG,SAA7zG,GAAw0G,SAAx0G,GAAm1G,SAAn1G,GAA81G,SAA91G,GAAy2G,SAAz2G,GAAo3G,SAAp3G,GAA+3G,SAA/3G,GAA04G,SAA14G,GAAq5G,SAAr5G,GAAg6G,SAAh6G,GAA26G,SAA36G,GAAs7G,SAAt7G,GAAi8G,SAAj8G,GAA48G,SAA58G,GAAu9G,SAAv9G,GAAk+G,SAAl+G,GAA6+G,SAA7+G,GAAw/G,SAAx/G,GAAmgH,SAAngH,GAA8gH,SAA9gH,GAAyhH,SAAzhH,GAAoiH,SAApiH,GAA+iH,SAA/iH,GAA0jH,SAA1jH,GAAqkH,SAArkH,GAAglH,SAAhlH,GAA2lH,SAA3lH,GAAsmH,SAAtmH,GAAinH,SAAjnH,GAA4nH,SAA5nH,GAAuoH,SAAvoH,GAAkpH,SAAlpH,GAA6pH,SAA7pH,GAAwqH,SAAxqH,GAAmrH,SAAnrH,GAA8rH,SAA9rH,GAAysH,SAAzsH,GAAotH,SAAptH,GAA+tH,SAA/tH,GAA0uH,SAA1uH,GAAqvH,SAArvH,GAAgwH,SAAhwH,GAA2wH,SAA3wH,GAAsxH,SAAtxH,GAAiyH,SAAjyH,GAA4yH,SAA5yH,GAAuzH,SAAvzH,GAAk0H,SAAl0H,GAA60H,SAA70H,GAAw1H,SAAx1H,GAAm2H,SAAn2H,GAA82H,SAA92H,GAAy3H,SAAz3H,GAAo4H,SAAp4H,GAA+4H,SAA/4H,GAA05H,SAA15H,GAAq6H,SAAr6H,GAAg7H,SAAh7H,GAA27H,SAA37H,GAAs8H,SAAt8H,GAAi9H,SAAj9H,GAA49H,SAA59H,GAAu+H,SAAv+H,GAAk/H,SAAl/H,GAA6/H,SAA7/H,GAAwgI,SAAxgI,GAAmhI,SAAnhI,GAA8hI,SAA9hI,GAAyiI,SAAziI,GAAojI,SAApjI,GAA+jI,SAA/jI,GAA0kI,SAA1kI,GAAqlI,SAArlI,GAAgmI,SAAhmI,GAA2mI,SAA3mI,GAAsnI,SAAtnI,GAAioI,SAAjoI,GAA4oI,SAA5oI,GAAupI,SAAvpI,GAAkqI,SAAlqI,GAA6qI,SAA7qI,GAAwrI,SAAxrI,GAAmsI,SAAnsI,GAA8sI,SAA9sI,GAAytI,SAAztI,GAAouI,SAApuI,GAA+uI,SAA/uI,GAA0vI,SAA1vI,GAAqwI,SAArwI,GAAgxI,SAAhxI,GAA2xI,SAA3xI,GAAsyI,SAAtyI,GAAizI,SAAjzI,GAA4zI,SAA5zI,GAAu0I,SAAv0I,GAAk1I,SAAl1I,GAA61I,SAA71I,GAAw2I,SAAx2I,GAAm3I,SAAn3I,GAA83I,SAA93I,GAAy4I,SAAz4I,GAAo5I,SAAp5I,GAA+5I,SAA/5I,GAA06I,SAA16I,GAAq7I,SAAr7I,GAAg8I,SAAh8I,GAA28I,SAA38I,GAAs9I,SAAt9I,GAAi+I,SAAj+I,GAA4+I,SAA5+I,GAAu/I,SAAv/I,GAAkgJ,SAAlgJ,GAA6gJ,SAA7gJ,GAAwhJ,SAAxhJ,GAAmiJ,SAAniJ,GAA8iJ,SAA9iJ,GAAyjJ,SAAzjJ,GAAokJ,SAApkJ,GAA+kJ,SAA/kJ,GAA0lJ,SAA1lJ,GAAqmJ,SAArmJ,GAAgnJ,SAAhnJ,GAA2nJ,SAA3nJ,GAAsoJ,SAAtoJ,GAAipJ,SAAjpJ,GAA4pJ,SAA5pJ,GAAuqJ,SAAvqJ,GAAkrJ,SAAlrJ,GAA6rJ,SAA7rJ,GAAwsJ,SAAxsJ,GAAmtJ,SAAntJ,GAA8tJ,SAA9tJ,GAAyuJ,SAAzuJ,GAAovJ,SAApvJ,GAA+vJ,SAA/vJ,GAA0wJ,SAA1wJ,GAAqxJ,SAArxJ,GAAgyJ,SAAhyJ,GAA2yJ,SAA3yJ,GAAszJ,SAAtzJ,GAAi0J,SAAj0J,GAA40J,SAA50J,GAAu1J,SAAv1J,GAAk2J,SAAl2J,GAA62J,SAA72J,GAAw3J,SAAx3J,GAAm4J,SAAn4J,GAA84J,SAA94J,GAAy5J,SAAz5J,GAAo6J,SAAp6J,GAA+6J,SAA/6J,GAA07J,SAA17J,GAAq8J,SAAr8J,GAAg9J,SAAh9J,GAA29J,SAA39J,GAAs+J,SAAt+J,GAAi/J,SAAj/J,GAA4/J,SAA5/J,GAAugK,SAAvgK,GAAkhK,SAAlhK,GAA6hK,SAA7hK,GAAwiK,SAAxiK,GAAmjK,SAAnjK,GAA8jK,SAA9jK,GAAykK,SAAzkK,GAAolK,SAAplK,GAA+lK,SAA/lK,GAA0mK,SAA1mK,GAAqnK,SAArnK,GAAgoK,SAAhoK,GAA2oK,SAA3oK,GAAspK,SAAtpK,GAAiqK,SAAjqK,GAA4qK,SAA5qK,GAAurK,SAAvrK,GAAksK,SAAlsK,GAA6sK,SAA7sK,GAAwtK,SAAxtK,GAAmuK,SAAnuK,GAA8uK,SAA9uK,GAAyvK,SAAzvK,GAAowK,SAApwK,GAA+wK,SAA/wK,GAA0xK,SAA1xK,GAAqyK,SAAryK,GAAgzK,SAAhzK,GAA2zK,SAA3zK,GAAs0K,SAAt0K,GAAi1K,SAAj1K,GAA41K,SAA51K,GAAu2K,SAAv2K,GAAk3K,SAAl3K,GAA63K,SAA73K,GAAw4K,SAAx4K,GAAm5K,SAAn5K,GAA85K,SAA95K,GAAy6K,SAAz6K,GAAo7K,SAAp7K,GAA+7K,SAA/7K,GAA08K,SAA18K,GAAq9K,SAAr9K,GAAg+K,SAAh+K,GAA2+K,SAA3+K,GAAs/K,SAAt/K,GAAigL,SAAjgL,GAA4gL,SAA5gL,GAAuhL,SAAvhL,GAAkiL,SAAliL,GAA6iL,SAA7iL,GAAwjL,SAAxjL,GAAmkL,SAAnkL,GAA8kL,SAA9kL,GAAylL,SAAzlL,GAAomL,SAApmL,GAA+mL,SAA/mL,GAA0nL,SAA1nL,GAAqoL,SAAroL,GAAgpL,SAAhpL,GAA2pL,SAA3pL,GAAsqL,SAAtqL,GAAirL,SAAjrL,GAA4rL,SAA5rL,GAAusL,SAAvsL,GAAktL,SAAltL,GAA6tL,SAA7tL,GAAwuL,SAAxuL,GAAmvL,SAAnvL,GAA8vL,SAA9vL,GAAywL,SAAzwL,GAAoxL,SAApxL,GAA+xL,SAA/xL,GAA0yL,SAA1yL,GAAqzL,SAArzL,GAAg0L,SAAh0L,GAA20L,SAA30L,GAAs1L,SAAt1L,GAAi2L,SAAj2L,GAA42L,SAA52L,GAAu3L,SAAv3L,GAAk4L,SAAl4L,GAA64L,SAA74L,GAAw5L,SAAx5L,GAAm6L,SAAn6L,GAA86L,SAA96L,GAAy7L,SAAz7L,GAAo8L,SAAp8L,GAA+8L,SAA/8L,GAA09L,SAA19L,GAAq+L,SAAr+L,GAAg/L,SAAh/L,GAA2/L,SAA3/L,GAAsgM,SAAtgM,GAAihM,SAAjhM,GAA4hM,SAA5hM,GAAuiM,SAAviM,GAAkjM,SAAljM,GAA6jM,SAA7jM,GAAwkM,SAAxkM,GAAmlM,SAAnlM,GAA8lM,SAA9lM,GAAymM,SAAzmM,GAAonM,SAApnM,GAA+nM,SAA/nM,GAA0oM,SAA1oM,GAAqpM,SAArpM,GAAgqM,SAAhqM,GAA2qM,SAA3qM,GAAsrM,SAAtrM,GAAisM,SAAjsM,GAA4sM,SAA5sM,GAAutM,SAAvtM,GAAkuM,SAAluM,GAA6uM,SAA7uM,GAAwvM,SAAxvM,GAAmwM,SAAnwM,GAA8wM,SAA9wM,GAAyxM,SAAzxM,GAAoyM,SAApyM,GAA+yM,SAA/yM,GAA0zM,SAA1zM,GAAq0M,SAAr0M,GAAg1M,SAAh1M,GAA21M,SAA31M,GAAs2M,SAAt2M,GAAi3M,SAAj3M,GAA43M,SAA53M,GAAu4M,SAAv4M,GAAk5M,SAAl5M,GAA65M,SAA75M,GAAw6M,SAAx6M,GAAm7M,SAAn7M,GAA87M,SAA97M,GAAy8M,SAAz8M,GAAo9M,SAAp9M,GAA+9M,SAA/9M,GAA0+M,SAA1+M,GAAq/M,SAAr/M,GAAggN,SAAhgN,GAA2gN,SAA3gN,GAAshN,SAAthN,GAAiiN,SAAjiN,GAA4iN,SAA5iN,GAAujN,SAAvjN,GAAkkN,SAAlkN,GAA6kN,SAA7kN,GAAwlN,SAAxlN,GAAmmN,SAAnmN,GAA8mN,SAA9mN,GAAynN,SAAznN,GAAooN,SAApoN,GAA+oN,SAA/oN,GAA0pN,SAA1pN,GAAqqN,SAArqN,GAAgrN,SAAhrN,GAA2rN,SAA3rN,GAAssN,SAAtsN,GAAitN,SAAjtN,GAA4tN,SAA5tN,GAAuuN,SAAvuN,GAAkvN,SAAlvN,GAA6vN,SAA7vN,GAAwwN,SAAxwN,GAAmxN,SAAnxN,GAA8xN,SAA9xN,GAAyyN,SAAzyN,GAAozN,SAApzN,GAA+zN,SAA/zN,GAA00N,SAA10N,GAAq1N,SAAr1N,GAAg2N,SAAh2N,GAA22N,SAA32N,GAAs3N,SAAt3N,GAAi4N,SAAj4N,GAA44N,SAA54N,GAAu5N,SAAv5N,GAAk6N,SAAl6N,GAA66N,SAA76N,GAAw7N,SAAx7N,GAAm8N,SAAn8N,GAA88N,SAA98N,GAAy9N,SAAz9N,GAAo+N,SAAp+N,GAA++N,SAA/+N,GAA0/N,SAA1/N,GAAqgO,SAArgO,GAAghO,SAAhhO,GAA2hO,SAA3hO,GAAsiO,SAAtiO,GAAijO,SAAjjO,GAA4jO,SAA5jO,GAAukO,SAAvkO,GAAklO,SAAllO,GAA6lO,SAA7lO,GAAwmO,SAAxmO,GAAmnO,SAAnnO,GAA8nO,SAA9nO,GAAyoO,SAAzoO,GAAopO,SAAppO,GAA+pO,SAA/pO,GAA0qO,SAA1qO,GAAqrO,SAArrO,GAAgsO,SAAhsO,GAA2sO,SAA3sO,GAAstO,SAAttO,GAAiuO,SAAjuO,GAA4uO,SAA5uO,GAAuvO,SAAvvO,GAAkwO,SAAlwO,GAA6wO,SAA7wO,GAAwxO,SAAxxO,GAAmyO,SAAnyO,GAA8yO,SAA9yO,GAAyzO,SAAzzO,GAAo0O,SAAp0O,GAA+0O,SAA/0O,GAA01O,SAA11O,GAAq2O,SAAr2O,GAAg3O,SAAh3O,GAA23O,SAA33O,GAAs4O,SAAt4O,GAAi5O,SAAj5O,GAA45O,SAA55O,GAAu6O,SAAv6O,GAAk7O,SAAl7O,GAA67O,SAA77O,GAAw8O,SAAx8O,GAAm9O,SAAn9O,GAA89O,SAA99O,GAAy+O,SAAz+O,GAAo/O,SAAp/O,GAA+/O,SAA//O,GAA0gP,SAA1gP,GAAqhP,SAArhP,GAAgiP,SAAhiP,GAA2iP,SAA3iP,GAAsjP,SAAtjP,GAAikP,SAAjkP,GAA4kP,SAA5kP,GAAulP,SAAvlP,GAAkmP,SAAlmP,GAA6mP,SAA7mP,GAAwnP,SAAxnP,GAAmoP,SAAnoP,GAA8oP,SAA9oP,GAAypP,SAAzpP,GAAoqP,SAApqP,GAA+qP,SAA/qP,GAA0rP,SAA1rP,GAAqsP,SAArsP,GAAgtP,SAAhtP,GAA2tP,SAA3tP,GAAsuP,SAAtuP,GAAivP,SAAjvP,GAA4vP,SAA5vP,GAAuwP,SAAvwP,GAAkxP,SAAlxP,GAA6xP,SAA7xP,GAAwyP,SAAxyP,GAAmzP,SAAnzP,GAA8zP,SAA9zP,GAAy0P,SAAz0P,GAAo1P,SAAp1P,GAA+1P,SAA/1P,GAA02P,SAA12P,GAAq3P,SAAr3P,GAAg4P,SAAh4P,GAA24P,SAA34P,GAAs5P,SAAt5P,GAAi6P,SAAj6P,GAA46P,SAA56P,GAAu7P,SAAv7P,GAAk8P,SAAl8P,GAA68P,SAA78P,GAAw9P,SAAx9P,GAAm+P,SAAn+P,GAA8+P,SAA9+P,GAAy/P,SAAz/P,GAAogQ,SAApgQ,GAA+gQ,SAA/gQ,GAA0hQ,SAA1hQ,GAAqiQ,SAAriQ,GAAgjQ,SAAhjQ,GAA2jQ,SAA3jQ,GAAskQ,SAAtkQ,GAAilQ,SAAjlQ,GAA4lQ,SAA5lQ,GAAumQ,SAAvmQ,GAAknQ,SAAlnQ,GAA6nQ,SAA7nQ,GAAwoQ,SAAxoQ,GAAmpQ,SAAnpQ,GAA8pQ,SAA9pQ,GAAyqQ,SAAzqQ,GAAorQ,SAAprQ,GAA+rQ,SAA/rQ,GAA0sQ,SAA1sQ,GAAqtQ,SAArtQ,GAAguQ,SAAhuQ,GAA2uQ,SAA3uQ,GAAsvQ,SAAtvQ,GAAiwQ,SAAjwQ,GAA4wQ,SAA5wQ,GAAuxQ,SAAvxQ,GAAkyQ,SAAlyQ,GAA6yQ,SAA7yQ,GAAwzQ,SAAxzQ,GAAm0Q,SAAn0Q,GAA80Q,SAA90Q,GAAy1Q,SAAz1Q,GAAo2Q,SAAp2Q,GAA+2Q,SAA/2Q,GAA03Q,SAA13Q,GAAq4Q,SAAr4Q,GAAg5Q,SAAh5Q,GAA25Q,SAA35Q,GAAs6Q,SAAt6Q,GAAi7Q,SAAj7Q,GAA47Q,SAA57Q,GAAu8Q,SAAv8Q,GAAk9Q,SAAl9Q,GAA69Q,SAA79Q,GAAw+Q,SAAx+Q,GAAm/Q,SAAn/Q,GAA8/Q,SAA9/Q,GAAygR,SAAzgR,GAAohR,SAAphR,GAA+hR,SAA/hR,GAA0iR,SAA1iR,GAAqjR,SAArjR,GAAgkR,SAAhkR,GAA2kR,SAA3kR,GAAslR,SAAtlR,GAAimR,SAAjmR,GAA4mR,SAA5mR,GAAunR,SAAvnR,GAAkoR,SAAloR,GAA6oR,SAA7oR,GAAwpR,SAAxpR,GAAmqR,SAAnqR,GAA8qR,SAA9qR,GAAyrR,SAAzrR,GAAosR,SAApsR,GAA+sR,SAA/sR,GAA0tR,SAA1tR,GAAquR,SAAruR,GAAgvR,SAAhvR,GAA2vR,SAA3vR,GAAswR,SAAtwR,GAAixR,SAAjxR,GAA4xR,SAA5xR,GAAuyR,SAAvyR,GAAkzR,SAAlzR,GAA6zR,SAA7zR,GAAw0R,SAAx0R,GAAm1R,SAAn1R,GAA81R,SAA91R,GAAy2R,SAAz2R,GAAo3R,SAAp3R,GAA+3R,SAA/3R,GAA04R,SAA14R,GAAq5R,SAAr5R,GAAg6R,SAAh6R,GAA26R,SAA36R,GAAs7R,SAAt7R,GAAi8R,SAAj8R,GAA48R,SAA58R,GAAu9R,SAAv9R,GAAk+R,SAAl+R,GAA6+R,SAA7+R,GAAw/R,SAAx/R,GAAmgS,SAAngS,GAA8gS,SAA9gS,GAAyhS,SAAzhS,GAAoiS,SAApiS,GAA+iS,SAA/iS,GAA0jS,SAA1jS,GAAqkS,SAArkS,GAAglS,SAAhlS,GAA2lS,SAA3lS,GAAsmS,SAAtmS,GAAinS,SAAjnS,GAA4nS,SAA5nS,GAAuoS,SAAvoS,GAAkpS,SAAlpS,GAA6pS,SAA7pS,GAAwqS,SAAxqS,GAAmrS,SAAnrS,GAA8rS,SAA9rS,GAAysS,SAAzsS,GAAotS,SAAptS,GAA+tS,SAA/tS,GAA0uS,SAA1uS,GAAqvS,SAArvS,GAAgwS,SAAhwS,GAA2wS,SAA3wS,GAAsxS,SAAtxS,GAAiyS,SAAjyS,GAA4yS,SAA5yS,GAAuzS,SAAvzS,GAAk0S,SAAl0S,GAA60S,SAA70S,GAAw1S,SAAx1S,GAAm2S,SAAn2S,GAA82S,SAA92S,GAAy3S,SAAz3S,GAAo4S,SAAp4S,GAA+4S,SAA/4S,GAA05S,SAA15S,GAAq6S,SAAr6S,GAAg7S,SAAh7S,GAA27S,SAA37S,GAAs8S,SAAt8S,GAAi9S,SAAj9S,GAA49S,SAA59S,GAAu+S,SAAv+S,GAAk/S,SAAl/S,GAA6/S,SAA7/S,GAAwgT,SAAxgT,GAAmhT,SAAnhT,GAA8hT,SAA9hT,GAAyiT,SAAziT,GAAojT,SAApjT,GAA+jT,SAA/jT,GAA0kT,SAA1kT,GAAqlT,SAArlT,GAAgmT,SAAhmT,GAA2mT,SAA3mT,GAAsnT,SAAtnT,GAAioT,SAAjoT,GAA4oT,SAA5oT,GAAupT,SAAvpT,GAAkqT,SAAlqT,GAA6qT,SAA7qT,GAAwrT,SAAxrT,GAAmsT,SAAnsT,GAA8sT,SAA9sT,GAAytT,SAAztT,GAAouT,SAApuT,GAA+uT,SAA/uT,GAA0vT,SAA1vT,GAAqwT,SAArwT,GAAgxT,SAAhxT,GAA2xT,SAA3xT,GAAsyT,SAAtyT,GAAizT,SAAjzT,GAA4zT,SAA5zT,GAAu0T,SAAv0T,GAAk1T,SAAl1T,GAA61T,SAA71T,GAAw2T,SAAx2T,GAAm3T,SAAn3T,GAA83T,SAA93T,GAAy4T,SAAz4T,GAAo5T,SAAp5T,GAA+5T,SAA/5T,GAA06T,SAA16T,GAAq7T,SAAr7T,GAAg8T,SAAh8T,GAA28T,SAA38T,GAAs9T,SAAt9T,GAAi+T,SAAj+T,GAA4+T,SAA5+T,GAAu/T,SAAv/T,GAAkgU,SAAlgU,GAA6gU,SAA7gU,GAAwhU,SAAxhU,GAAmiU,SAAniU,GAA8iU,SAA9iU,GAAyjU,SAAzjU,GAAokU,SAApkU,GAA+kU,SAA/kU,GAA0lU,SAA1lU,GAAqmU,SAArmU,GAAgnU,SAAhnU,GAA2nU,SAA3nU,GAAsoU,SAAtoU,GAAipU,SAAjpU,GAA4pU,SAA5pU,GAAuqU,SAAvqU,GAAkrU,SAAlrU,GAA6rU,SAA7rU,GAAwsU,SAAxsU,GAAmtU,SAAntU,GAA8tU,SAA9tU,GAAyuU,SAAzuU,GAAovU,SAApvU,GAA+vU,SAA/vU,GAA0wU,SAA1wU,GAAqxU,SAArxU,GAAgyU,SAAhyU,GAA2yU,SAA3yU,GAAszU,SAAtzU,GAAi0U,SAAj0U,GAA40U,SAA50U,GAAu1U,SAAv1U,GAAk2U,SAAl2U,GAA62U,SAA72U,GAAw3U,SAAx3U,GAAm4U,SAAn4U,GAA84U,SAA94U,GAAy5U,SAAz5U,GAAo6U,SAAp6U,GAA+6U,SAA/6U,GAA07U,SAA17U,GAAq8U,SAAr8U,GAAg9U,SAAh9U,GAA29U,SAA39U,GAAs+U,SAAt+U,GAAi/U,SAAj/U,GAA4/U,SAA5/U,GAAugV,SAAvgV,GAAkhV,SAAlhV,GAA6hV,SAA7hV,GAAwiV,SAAxiV,GAAmjV,SAAnjV,GAA8jV,SAA9jV,GAAykV,SAAzkV,GAAolV,SAAplV,GAA+lV,SAA/lV,GAA0mV,SAA1mV,GAAqnV,SAArnV,GAAgoV,SAAhoV,GAA2oV,SAA3oV,GAAspV,SAAtpV,GAAiqV,SAAjqV,GAA4qV,SAA5qV,GAAurV,SAAvrV,GAAksV,SAAlsV,GAA6sV,SAA7sV,GAAwtV,SAAxtV,GAAmuV,SAAnuV,GAA8uV,SAA9uV,GAAyvV,SAAzvV,GAAowV,SAApwV,GAA+wV,SAA/wV,GAA0xV,SAA1xV,GAAqyV,SAAryV,GAAgzV,SAAhzV,GAA2zV,SAA3zV,GAAs0V,SAAt0V,GAAi1V,SAAj1V,GAA41V,SAA51V,GAAu2V,SAAv2V,GAAk3V,SAAl3V,GAA63V,SAA73V,GAAw4V,SAAx4V,GAAm5V,SAAn5V,GAA85V,SAA95V,GAAy6V,SAAz6V,GAAo7V,SAAp7V,GAA+7V,SAA/7V,GAA08V,SAA18V,GAAq9V,SAAr9V,GAAg+V,SAAh+V,GAA2+V,S

AAA,6B;MAAA,sC;K;;IAEA,uD;MAAA,6B;MAAA,+C;K;;IAEA,wD;MAAA,6B;MAAA,gD;K;;IAEA,oD;MAAA,6B;MAAA,4C;K;;IAEA,iD;MAAA,6B;MAAA,yC;K;;IAEA,wD;MAAA,6B;MAAA,gD;K;;IAEA,wD;MAAA,6B;MAAA,gD;K;;IAEA,6C;MAAA,6B;MAAA,qC;K;;IAEA,mD;MAAA,6B;MAAA,2C;K;;IAEA,6C;MAAA,6B;MAAA,qC;K;;IAEA,kD;MAAA,6B;MAAA,0C;K;;IAhCJ,mC;MAAA,+oB;K;;IAAA,wC;MAAA,a;aAAA,O;UAAA,2C;aAAA,kB;UAAA,sD;aAAA,gB;UAAA,oD;aAAA,U;UAAA,8C;aAAA,O;UAAA,2C;aAAA,gB;UAAA,oD;aAAA,iB;UAAA,qD;aAAA,a;UAAA,iD;aAAA,U;UAAA,8C;aAAA,iB;UAAA,qD;aAAA,iB;UAAA,qD;aAAA,M;UAAA,0C;aAAA,Y;UAAA,gD;aAAA,M;UAAA,0C;aAAA,W;UAAA,+C;;UAAA,uE;;K;;IAqCA,4C;MAAA,e;MAAA,iB;MAAA,uB;K;IAAA,0C;MAAA,6C;O;MAMI,0E;MAEA,0E;MAEA,4E;K;;IAJA,kD;MAAA,gC;MAAA,0C;K;;IAEA,kD;MAAA,gC;MAAA,0C;K;;IAEA,mD;MAAA,gC;MAAA,2C;K;;IAVJ,sC;MAAA,sI;K;;IAAA,2C;MAAA,a;aAAA,Q;UAAA,+C;aAAA,Q;UAAA,+C;aAAA,S;UAAA,gD;;UAAA,0E;;K;;IAwB8B,gC;MAAC,oC;K;;IAAQE,0B;MAAC,qB;QAAA,iD;MAAA,kB;K;;IAAlC,sB;K;;IAAA,4B;K;;IC/EA,yB;K;;IAAQA,6B;K;;ICnBA,mB;MAEI,UAAU,IAAI,C;MACd,OAAW,OAAO,CAAX,GAAc,GAAd,GAAuB,MAAM,CAAN,I;K;IAAG,qB;MAAC,UAAU,SAAI,CAAJ,C;MAAK,OAAW,kBAAO,CAAX,GAAc,GAAd,GAAuB,QAAM,CAAN,C;K;IAAG,mC;MAAC,OAAO,IAAI,IAAI,CAAJ,EAAO,EAAO,CAAP,EAAO,IAAI,CAAJ,EAAO,EAAO,CAAP,EAAO,CAAJ,EAAO,IAAI,EAAA2B,CAAA3B,C;K;IAAG,qC;MAAC,OAAO,IAAI,IAAI,CAAJ,EAAO,EAAO,CAAP,EAAO,IAAI,CAAJ,EAAO,EAAO,CAAP,EAAO,CAAJ,EAAO,IAAI,EAAA2B,CAAA3B,C;K;IAAG,qD

;QAAA,OAAwB,mBAAM,iBAAN,EAAM,yBAAN,O;;QAAoB,MAAM,2BAAyB,UAAF,WAAvB,C;K;;IAJhF,kC;MACmD,iD;K;IAOpB,gC;MAAC,wB;K;;IAEhC,+B;MAC8C,MAAM,mC;K;IAEpD,8C;MAEI,IAAI,qBAAJ,C;QACI,OAAO,C5ByIiF,W4BzIrE,U5ByIqE,E4BzIzD,Q5ByIyD,C;;Q4BvIxF,OAAS,CAAY,qBAAsB,UAAtB,EAAkC,QAAlC,C;;K;IAI7B,2C;MAEI,IAAI,KAAY,kBAAhB,C;QAGI,KAAY,mBAAkB,QAAlB,C;;QAEH,QAAT,SAAA+C,CAAlB,IAAjC,KAAiC,C;IAIvD,sC;MAGwB,Q;MADpB,gBAAgB,IAAhB,KAAgB,E;MACI,IAAI,OCnGkB,ODmGT,OAAT,EAAqB,WAArB,CAAJ,C;QAChB,OAAI,aAAJ,GAAmB,KAAnB,WAAzB,GAAyC,I;;QAEzC,zC,c;;MAHJ,wB;MAKA,kBAAkB,K;MAClB,iBAAiB,W;MACjB,OAAO,S;K;IAIa,sB;MAAC,U;K;iCACrB,iB;MACI,OAAO,mCAAsB,WAAtB,KAAmC,E;K;mCAG5C,Y;MACI,OAAO,M;K;mCAGX,Y;MACI,OAAuC,oBAAnB,BAAmB,UAAnB,EAAmB,GAAN,K;K;kCAGpC,Y;MAEI,OAAO,M;K;;+DAIf,gB;MAEI,YAAY,MAAY,IAAK,OAAjB,jB,C;MACZ,sBAAsB,IAAtB,EAAsB,a;QAClB,IAAI,UAAU,KAAK,CAAL,CAAV,CAAJ,C;UACI,OAAO,K;;;MAGf,OAAO,I;K;IAGX,qB;MAEI,OAAO,CAAC,KAAK,CAAL,CAAR,C;K;IAGJ

Y,OAAjB,IAA2B,CAA/B,C;QAAA,OAES,SAAL,CAAiB,QAAjB,EAA2B,UAA3B,EAAuC,IAAvC,C;;QA0B0B,Q;QA5B9B,4DAImD,sEAJnD,EA4B8B,qBAxBS,UAwBT,qCA5B9B,C;;K;IASJ,gC;MAWK,kBAAD,M;MAAA,kBAAC,qEAAD,4DAA2C,S;K;6CAG/C,yB;MAAA,mG;MAAA,yH;MAAA,6F;QAKW,kC;QAAS,4C;O;MALpB,sEAMQ,Y;QACI,Q;QAAA,sC;UAAiB,U;;QACjB,OAAO,oB;O;MARnB,6G;MAAA,oC;QAKkC,Q;QAA9B,mEAA8B,oEAA9B,C;O;KALJ,C;iFC7HA,a;MAC6C,OAAA,MAAa,YAAW,CAAX,C;K;ICM3B,iC;;MAA6E,Q;MAAA,+BAAS,I;sCAAlB,lB,O,2DAAA,O;;;;K;;;;;;;;IAC/F,2B;MAAA,iD;MAAuB,oBAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sC;MAAA,iD;MAAuC,oBAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oC;MAAA,iD;MAAwC,oBAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG8C,kD;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACxG,8C;MAAA,kE;MAAuB,qCAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,yD;MAAA,kE;MAAuC,qCAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,uD;MAAA,kE;MAAwC,qCAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG2C,+C;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACrG,2C;MAAA,+D;MAAuB,kCAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sD;MAAA,+D;MAAuC,kCAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oD;MAAA,+D;MAAwC,kCAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG+C,6C;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC7G,yC;MAAA,6D;MAAuB,gCAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,oD;MAAA,6D;MAAuC,gCAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,kD;MAAA,6D;MAAwC,gCAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGiD,+C;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC/G,2C;MAAA,+D;MAAuB,kCAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sD;MAAA,+D;MAAuC,kCAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oD;MAAA,+D;MAAwC,kCAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGmD,iD;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACjH,6C;MAAA,iE;MAAuB,oCAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,wD;MAAA,iE;MAAuC,oCAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,sD;MAAA,iE;MAAwC,oCAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGqD,mD;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACnH,+C;MAAA,mE;MAAuB,sCAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,0D;MAAA,mE;MAAuC,sCAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,wD;MAAA,mE;MAAwC,sCAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGuD,qD;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACrH,iD;MAAA,qE;MAAuB,wCAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,4D;MAAA,qE;MAAuC,wCAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,0D;MAAA,qE;MAAwC,wCAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGyD,uD;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACvH,mD;MAAA,uE;MAAuB,0CAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,8D;MAAA,uE;MAAuC,0CAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,4D;MAAA,uE;MAAwC,0CAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG2D,yD;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACzH,qD;MAAA,yE;MAAuB,4CAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,gE;MAAA,yE;MAAuC,4CAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,8D;MAAA,yE;MAAwC,4CAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG6D,2D;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC3H,uD;MAAA,2E;MAAuB,8CAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,kE;MAAA,2E;MAAuC,8CAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,gE;MAAA,2E;MAAwC,8CAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG+D,6D;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC7H,yD;MAAA,6E;MAAuB,gDAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,oE;MAAA,6E;MAAuC,gDAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,kE;MAAA,6E;MAAwC,gDAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGiE,+D;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC/H,2D;MAAA,+E;MAAuB,kDAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sE;MAAA,+E;MAAuC,kDAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oE;MAAA,+E;MAAwC,kDAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGmE,iE;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACjI,6D;MAAA,iF;MAAuB,oDAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,wE;MAAA,iF;MAAuC,oDAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,sE;MAAA,iF;MAAwC,oDAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGqE,mE;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACnI,+D;MAAA,mF;MAAuB,sDAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,0E;MAAA,mF;MAAuC,sDAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,wE;MAAA,mF;MAAwC,sDAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGuE,qE;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACrI,iE;MAAA,qF;MAAuB,wDAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,4E;MAAA,qF;MAAuC,wDAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,0E;MAAA,qF;MAAwC,wDAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGyE,uE;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACvI,mE;MAAA,uF;MAAuB,0DAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,8E;MAAA,uF;MAAuC,0DAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,4E;MAAA,uF;MAAwC,0DAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG2E,yE;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACzI,qE;MAAA,yF;MAAuB,4DAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,gF;MAAA,yF;MAAuC,4DAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,8E;MAAA,yF;MAAwC,4DAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG6E,2E;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC3I,uE;MAAA,2F;MAAuB,8DAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,kF;MAAA,2F;MAAuC,8DAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,gF;MAAA,2F;MAAwC,8DAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG+E,6E;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC7I,yE;MAAA,6F;MAAuB,gEAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,oF;MAAA,6F;MAAuC,gEAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,kF;MAAA,6F;MAAwC,gEAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGiF,+E;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC/I,2E;MAAA,+F;MAAuB,kEAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sF;MAAA,+F;MAAuC,kEAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oF;MAAA,+F;MAAwC,kEAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGmF,iF;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACjJ,6E;MAAA,iG;MAAuB,oEAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,wF;MAAA,iG;MAAuC,oEAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,sF;MAAA,iG;MAAwC,oEAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGqF,mF;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACnJ,+E;MAAA,mG;MAAuB,sEAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,0F;MAAA,mG;MAAuC,sEAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,wF;MAAA,mG;MAAwC,sEAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGuF,qF;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACrJ,iF;MAAA,qG;MAAuB,wEAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,4F;MAAA,qG;MAAuC,wEAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,0F;MAAA,qG;MAAwC,wEAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGyF,uF;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACvJ,mF;MAAA,uG;MAAuB,0EAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,8F;MAAA,uG;MAAuC,0EAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,4F;MAAA,uG;MAAwC,0EAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG2F,yF;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACzJ,qF;MAAA,yG;MAAuB,4EAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,gG;MAAA,yG;MAAuC,4EAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,8F;MAAA,yG;MAAwC,4EAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG6F,2F;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC3J,uF;MAAA,2G;MAAuB,8EAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,kG;MAAA,2G;MAAuC,8EAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,gG;MAAA,2G;MAAwC,8EAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG+F,6F;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC7J,yF;MAAA,6G;MAAuB,gFAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,oG;MAAA,6G;MAAuC,gFAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,kG;MAAA,6G;MAAwC,gFAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGiG,+F;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC/J,2F;MAAA,+G;MAAuB,kFAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sG;MAAA,+G;MAAuC,kFAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oG;MAAA,+G;MAAwC,kFAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGmG,iG;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACjK,6F;MAAA,iH;MAAuB,oFAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,wG;MAAA,iH;MAAuC,oFAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,sG;MAAA,iH;MAAwC,oFAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGqG,mG;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACnK,+F;MAAA,mH;MAAuB,sFAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,0G;MAAA,mH;MAAuC,sFAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,wG;MAAA,mH;MAAwC,sFAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGuG,qG;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACrK,iG;MAAA,qH;MAAuB,wFAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,4G;MAAA,qH;MAAuC,wFAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,0G;MAAA,qH;MAAwC,wFAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGyG,uG;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACvK,mG;MAAA,uH;MAAuB,0FAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,8G;MAAA,uH;MAAuC,0FAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,4G;MAAA,uH;MAAwC,0FAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG2G,yG;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACzK,qG;MAAA,yH;MAAuB,4FAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,gH;MAAA,yH;MAAuC,4FAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,8G;MAAA,yH;MAAwC,4FAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG6G,2G;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC3K,uG;MAAA,2H;MAAuB,8FAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,kH;MAAA,2H;MAAuC,8FAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,gH;MAAA,2H;MAAwC,8FAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG+G,6G;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC7K,yG;MAAA,6H;MAAuB,gGAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,oH;MAAA,6H;MAAuC,gGAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,kH;MAAA,6H;MAAwC,gGAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGiH,+G;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC/K,2G;MAAA,+H;MAAuB,kGAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sH;MAAA,+H;MAAuC,kGAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oH;MAAA,+H;MAAwC,kGAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGmH,iH;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACjL,6G;MAAA,iI;MAAuB,oGAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,wH;MAAA,iI;MAAuC,oGAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,sH;MAAA,iI;MAAwC,oGAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGqH,mH;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACnL,+G;MAAA,mI;MAAuB,sGAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,0H;MAAA,mI;MAAuC,sGAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,wH;MAAA,mI;MAAwC,sGAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGuH,qH;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACrL,iH;MAAA,qI;MAAuB,wGAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,4H;MAAA,qI;MAAuC,wGAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,0H;MAAA,qI;MAAwC,wGAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGyH,uH;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACvL,mH;MAAA,uI;MAAuB,0GAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,8H;MAAA,uI;MAAuC,0GAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,4H;MAAA,uI;MAAwC,0GAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG2H,yH;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACzL,qH;MAAA,yI;MAAuB,4GAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,gI;MAAA,yI;MAAuC,4GAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,8H;MAAA,yI;MAAwC,4GAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG6H,2H;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC3L,uH;MAAA,2I;MAAuB,8GAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,kI;MAAA,2I;MAAuC,8GAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,gI;MAAA,2I;MAAwC,8GAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG+H,6H;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC7L,yH;MAAA,6I;MAAuB,gHAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,oI;MAAA,6I;MAAuC,gHAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,kI;MAAA,6I;MAAwC,gHAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGiI,+H;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC/L,2H;MAAA,+I;MAAuB,kHAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sI;MAAA,+I;MAAuC,kHAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oI;MAAA,+I;MAAwC,kHAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGmI,iI;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACjM,6H;MAAA,iJ;MAAuB,oHAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,wI;MAAA,iJ;MAAuC,oHAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,sI;MAAA,iJ;MAAwC,oHAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGqI,mI;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACnM,+H;MAAA,mJ;MAAuB,sHAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,0I;MAAA,mJ;MAAuC,sHAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,wI;MAAA,mJ;MAAwC,sHAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGuI,qI;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACrM,iI;MAAA,qJ;MAAuB,wHAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,4I;MAAA,qJ;MAAuC,wHAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,0I;MAAA,qJ;MAAwC,wHAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGyI,uI;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACvM,mI;MAAA,uJ;MAAuB,0HAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,8I;MAAA,uJ;MAAuC,0HAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,4I;MAAA,uJ;MAAwC,0HAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG2I,yI;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACzM,qI;MAAA,yJ;MAAuB,4HAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,gJ;MAAA,yJ;MAAuC,4HAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,8I;MAAA,yJ;MAAwC,4HAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG6I,2I;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC3M,uI;MAAA,2J;MAAuB,8HAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,kJ;MAAA,2J;MAAuC,8HAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,gJ;MAAA,2J;MAAwC,8HAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG+I,6I;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC7M,yI;MAAA,6J;MAAuB,gIAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,oJ;MAAA,6J;MAAuC,gIAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,kJ;MAAA,6J;MAAwC,gIAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGiJ,+I;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC/M,2I;MAAA,+J;MAAuB,kIAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sJ;MAAA,+J;MAAuC,kIAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oJ;MAAA,+J;MAAwC,kIAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGmJ,iJ;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACjN,6I;MAAA,iK;MAAuB,oIAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,wJ;MAAA,iK;MAAuC,oIAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,sJ;MAAA,iK;MAAwC,oIAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGqJ,mJ;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACnN,+I;MAAA,mK;MAAuB,sIAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,0J;MAAA,mK;MAAuC,sIAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,wJ;MAAA,mK;MAAwC,sIAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGuJ,qJ;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACrN,iJ;MAAA,qK;MAAuB,wIAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,4J;MAAA,qK;MAAuC,wIAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,0J;MAAA,qK;MAAwC,wIAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGyJ,uJ;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACvN,mJ;MAAA,uK;MAAuB,0IAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,8J;MAAA,uK;MAAuC,0IAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,4J;MAAA,uK;MAAwC,0IAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG2J,yJ;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACzN,qJ;MAAA,yK;MAAuB,4IAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,gK;MAAA,yK;MAAuC,4IAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,8J;MAAA,yK;MAAwC,4IAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG6J,2J;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC3N,uJ;MAAA,2K;MAAuB,8IAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,kK;MAAA,2K;MAAuC,8IAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,gK;MAAA,2K;MAAwC,8IAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG+J,6J;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC7N,yJ;MAAA,6K;MAAuB,gJAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,oK;MAAA,6K;MAAuC,gJAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,kK;MAAA,6K;MAAwC,gJAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGiK,+J;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC/N,2J;MAAA,+K;MAAuB,kJAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sK;MAAA,+K;MAAuC,kJAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oK;MAAA,+K;MAAwC,kJAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGmK,iK;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACjO,6J;MAAA,iL;MAAuB,oJAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,wK;MAAA,iL;MAAuC,oJAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,sK;MAAA,iL;MAAwC,oJAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGqK,mK;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACnO,+J;MAAA,mL;MAAuB,sJAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,0K;MAAA,mL;MAAuC,sJAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,wK;MAAA,mL;MAAwC,sJAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGuK,qK;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACrO,iK;MAAA,qL;MAAuB,wJAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,4K;MAAA,qL;MAAuC,wJAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,0K;MAAA,qL;MAAwC,wJAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGyK,uK;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACvO,mK;MAAA,uL;MAAuB,0JAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,8K;MAAA,uL;MAAuC,0JAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,4K;MAAA,uL;MAAwC,0JAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG2K,yK;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACzO,qK;MAAA,yL;MAAuB,4JAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,gL;MAAA,yL;MAAuC,4JAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,8K;MAAA,yL;MAAwC,4JAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG6K,2K;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC3O,uK;MAAA,2L;MAAuB,8JAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,kL;MAAA,2L;MAAuC,8JAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,gL;MAAA,2L;MAAwC,8JAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAG+K,6K;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC7O,yK;MAAA,6L;MAAuB,gKAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,oL;MAAA,6L;MAAuC,gKAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,kL;MAAA,6L;MAAwC,gKAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGiL,+K;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IAC/O,2K;MAAA,+L;MAAuB,kKAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,sL;MAAA,+L;MAAuC,kKAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,oL;MAAA,+L;MAAwC,kKAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGmL,iL;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACjP,6K;MAAA,iM;MAAuB,oKAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,wL;MAAA,iM;MAAuC,oKAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,sL;MAAA,iM;MAAwC,oKAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGqL,mL;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACnP,+K;MAAA,mM;MAAuB,sKAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;IACA,0L;MAAA,mM;MAAuC,sKAAK,OAAL,EAAc,IAAd,C;MAAvC,Y;K;IACA,wL;MAAA,mM;MAAwC,sKAAK,SAAL,EAAgB,KAAhB,B,C;MAAxC,Y;K;IAGuL,qL;MAA0D,4BAAiB,OAAjB,EAA0B,EAA1B,B,C;;K;IACrP,iL;MAAA,qM;MAAuB,wKAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;K;

;oFAGX,oB;MACI,IAAI,IAAK,OAAL,KAAkB,SAAtB,C;QACI,YAAc,IAAK,O;;K;0EAI3B,wB;MAA+D,OAAA,MAAa,QAAO,GAAP,EAAY,OAAZ,C;K;IS/F5E,mC;MAOI,kBAAkB,MAAa,eAAc,SAAd,C;MAC/B,iBAAiB,MAAa,eAAc,IAAd,C;MAC9B,OAAW,gBAAe,UAAnB,GAA+B,SAA/B,GAAyC,CAAC,S;K;0ECUrD,2B;MAKyE,OAAA,MAAa,gBAAe,IAAf,C;K;4EAyBtF,2B;MAKsE,OAAA,MAAa,eAAc,IAAd,C;K;kEAGnF,qB;MACgD,OAAA,MAAa,KAAK,UAAS,GAAT,EAAc,IAAd,C;K;wEAChC,qB;MAAQ,OAAK,SAAY,a;K;0EACxB,qB;MAAQ,OAAK,SAAY,c;K;IC3D5D,0D;MAGI,OAAO,I;K;ICHX,sC;MAMsD,OAAA,MAAa,SAAY,UAAS,WAAW,KAAX,C;K;ItDKlE,uC;Mf2nBW,Q;MAAA,IAAe,nBgB,KfqnBZ,IAAS,CAAT,EAAY,IAAe,nBAAY,KfqnBvC,IAAS,wBAA3B,C;QAAA,OAAA,MAAa,IAAb,C;MAAA,OAAA,MAAa,IAAb,C;MAAA,OAAA,MAAa,IAAb,C;M

U,SAAV,EAAgB,WAAhB,EAA6B,iBAA7B,EAAgD,UAAhD,EAA4D,QAA5D,C;QACA,OAAO,W;O;KArBX,C;wFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA0C,WA1C,EAAiF,iBAAjF,EAAoG,UAApG,EAAgH,QAAhH,C;QACA,OAAO,W;O;KArBX,C;wFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA2C,WA3C,EAAmF,iBAAnF,EAAsG,UAAtG,EAAkH,QAAlH,C;QACA,OAAO,W;O;KArBX,C;wFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;uFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAA4C,WA5C,EAAqF,iBAArF,EAAwG,UAAxG,EAAoH,QAApH,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA6C,WA7C,EAAuF,iBAAvF,EAA0G,UAA1G,EAAsH,QAAtH,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA8C,WA9C,EAAyF,iBAAzF,EAA4G,UAA5G,EAAwH,QAAxH,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAA+C,WA/C,EAA2F,iBAA3F,EAA8G,UAA9G,EAA0H,QAA1H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAgD,WAhD,EAA4F,iBAA5F,EAA+G,UAA/G,EAA2H,QAA3H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAiD,WAjD,EAA6F,iBAA7F,EAAgH,UAAhH,EAA4H,QAA5H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAkD,WAlD,EAA8F,iBAA9F,EAAiH,UAAjH,EAA6H,QAA7H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAmD,WAnD,EAA+F,iBAA/F,EAAkH,UAAlH,EAA8H,QAA9H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAoD,WApD,EAAgG,iBAAhG,EAAmH,UAAnH,EAA+H,QAA/H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAqD,WArD,EAAiG,iBAAjG,EAAoH,UAApH,EAAgI,QAAhI,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAsD,WAtD,EAAkG,iBAAlG,EAAqH,UAArH,EAAiI,QAAjI,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAuD,WAvD,EAAmG,iBAAnG,EAAsH,UAAtH,EAAkI,QAAlI,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAwD,WAxD,EAAoG,iBAApG,EAAuH,UAAvH,EAAmI,QAAnI,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAyD,WAzD,EAAqG,iBAArG,EAAwH,UAAxH,EAAoI,QAApI,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA0D,WA1D,EAAsG,iBAAtG,EAAyH,UAAzH,EAAqI,QAArI,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAA2D,WA3D,EAAuG,iBAAvG,EAA0H,UAA1H,EAAsI,QAAtI,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA4D,WA5D,EAAwG,iBAAxG,EAA2H,UAA3H,EAAuI,QAAvI,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA6D,WA7D,EAAyG,iBAAzG,EAA4H,UAA5H,EAAwI,QAAxI,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAA8D,WA9D,EAA0G,iBAA1G,EAA6H,UAA7H,EAAyI,QAAzI,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA+D,WA/D,EAA2G,iBAA3G,EAA8H,UAA9H,EAA0I,QAA1I,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAgE,WAhE,EAA4G,iBAA5G,EAA+H,UAA/H,EAA2I,QAA3I,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAiE,WAnE,EAA6G,iBAA7G,EAAgI,UAAhI,EAA4I,QAA5I,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAkE,WApE,EAA8G,iBAA9G,EAAiI,UAAjI,EAA6I,QAA7I,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAmE,WArE,EAA+G,iBAA/G,EAAkI,UAAlI,EAA8I,QAA9I,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAoE,WAtE,EAAgH,iBAAhH,EAAmI,UAAnI,EAA+I,QAA/I,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAqE,WAvE,EAAiH,iBAAjH,EAAoI,UAApI,EAAgJ,QAAhJ,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAsE,WAxE,EAAkH,iBAAlH,EAAqI,UAArI,EAAiJ,QAAjJ,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAuE,WA3E,EAAmH,iBAAnH,EAAsI,UAAtI,EAAkJ,QAAlJ,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAwE,WA5E,EAAoH,iBAApH,EAAuI,UAAvI,EAAmJ,QAAnJ,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAyE,WA7E,EAAqH,iBAArH,EAAwI,UAAxI,EAAoJ,QAApJ,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAA0E,WAhF,EAAsH,iBAAtH,EAAyI,UAAzI,EAAqJ,QAArJ,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA2E,WAjF,EAAuH,iBAAvH,EAA0I,UAA1I,EAAsJ,QAAtJ,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA4E,WAlF,EAAwH,iBAAxH,EAA2I,UAA3I,EAAuJ,QAAvJ,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAA6E,WArF,EAAyH,iBAAzH,EAA4I,UAA5I,EAAwJ,QAAxJ,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA8E,WAtF,EAA0H,iBAA1H,EAA6I,UAA7I,EAAyJ,QAAzJ,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA+E,WAvF,EAA2H,iBAA3H,EAA8I,UAA9I,EAA0J,QAA1J,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAgF,WAzF,EAA4H,iBAA5H,EAA+I,UAA/I,EAA2J,QAA3J,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAiF,WA1F,EAA6H,iBAA7H,EAAgJ,UAAhJ,EAA4J,QAA5J,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAkF,WA3F,EAA8H,iBAA9H,EAAiJ,UAAjJ,EAA6J,QAA7J,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAmF,WA7F,EAA+H,iBAA/H,EAAkJ,UAAlJ,EAA8J,QAA9J,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAoF,WA9F,EAAgI,iBAAhI,EAAmJ,UAAnJ,EAA+J,QAA/J,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAqF,WA/F,EAAiI,iBAAjI,EAAoJ,UAApJ,EAAgK,QAAhK,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAsF,WAjG,EAAoI,iBAApI,EAAuJ,UAAvJ,EAAmK,QAAnK,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAuF,WAlG,EAAqI,iBAArI,EAAwJ,UAAxJ,EAAoK,QAApK,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAwF,WAnG,EAAsI,iBAAtI,EAAyJ,UAAzJ,EAAqK,QAArK,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAyF,WAtG,EAAuI,iBAAvI,EAA0J,UAA1J,EAAsK,QAAtK,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA0F,WAvG,EAAwI,iBAAxI,EAA2J,UAA3J,EAAuK,QAAvK,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA2F,WAxG,EAAyI,iBAAzI,EAA4J,UAA5J,EAAwK,QAAxK,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAA4F,WA1G,EAA0I,iBAA1I,EAA6J,UAA7J,EAAyK,QAAzK,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA6F,WA3G,EAA2I,iBAA3I,EAA8J,UAA9J,EAA0K,QAA1K,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA8F,WA5G,EAA4I,iBAA5I,EAA+J,UAA/J,EAA2K,QAA3K,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAA+F,WA9G,EAA6I,iBAA7I,EAAgK,UAAhK,EAA4K,QAA5K,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAgG,WA/G,EAA8I,iBAA9I,EAAiK,UAAjK,EAA6K,QAA7K,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAiG,WAhH,EAA+I,iBAA/I,EAAkK,UAAlK,EAA8K,QAA9K,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAkG,WAjH,EAAgJ,iBAAhJ,EAAmK,UAAnK,EAA+K,QAA/K,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAmG,WAlH,EAAiJ,iBAAjJ,EAAoK,UAApK,EAAgL,QAAhL,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAoG,WAnH,EAAkJ,iBAAlJ,EAAqK,UAArK,EAAiL,QAAjL,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAqG,WArH,EAAmJ,iBAAnJ,EAAsK,UAAtK,EAAkL,QAAlL,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAsG,WAtH,EAAoJ,iBAApJ,EAAuK,UAAvK,EAAmL,QAAnL,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAuG,WAvH,EAAqJ,iBAArJ,EAAwK,UAAxK,EAAoL,QAApL,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAwG,WAzH,EAAuJ,iBAAvJ,EAA0K,UAA1K,EAAsL,QAAtL,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAyG,WA1H,EAAwJ,iBAAxJ,EAA2K,UAA3K,EAAuL,QAAvL,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA0G,WA3H,EAAyJ,iBAAzJ,EAA4K,UAA5K,EAAwL,QAAxL,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAA2G,WA5H,EAA0J,iBAA1J,EAA6K,UAA7K,EAAyL,QAAzL,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA4G,WA7H,EAA2J,iBAA3J,EAA8K,UAA9K,EAA0L,QAA1L,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA6G,WA9H,EAA4J,iBAA5J,EAA+K,UAA/K,EAA2L,QAA3L,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAA8G,WA/H,EAA6J,iBAA7J,EAAgL,UAAhL,EAA4L,QAA5L,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA+G,WAhI,EAA8J,iBAA9J,EAAiL,UAAjL,EAA6L,QAA7L,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAAgH,WAjI,EAA+J,iBAA/J,EAAkL,UAAlL,EAA8L,QAA9L,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QAAC,H,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACrI,UAAU,SAAV,EAAiH,WAlI,EAAgK,iBAAhK,EAAmL,UAAnL,EAA+L,QAA/L,C;QACA,OAAO,W;O;KArBX,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAAkH,WAnI,EAAiK,iBAAjK,EAAoL,UAApL,

B,OwC7pCO,K;K;IxCgqCX,sD;MAUI,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;MACb,OA
AY,SAAY,OAAM,SAAN,EAAiB,OAAjB,C;K;IAG5B,sD;MAUI,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EAA
sC,gBAAtC,C;MACb,OAAY,SAAY,OAAM,SAAN,EAAiB,OAAjB,C;K;IAG5B,uD;MAUI,oCAAa,2BAAkB,SA
AlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;MACN,WAAS,c;MAAT,YAA8B,SAAY,OAAM,SAAN,EAAiB,OAAjB
,C;MwCxsCjD,eAAiB,I;MxCwsCjB,OwCvsCO,K;K;IxC0sCX,uD;MAUI,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7
B,EAAsC,gBAAtC,C;MACN,WAAS,W;MAAT,YAA2B,SAAY,OAAM,SAAN,EAAiB,OAAjB,C;MwCttC9C,eA
AiB,I;MxCstCjB,OwCrtCO,K;K;IxCwtCX,wD;MAWgD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MAC
/E,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EA
AyB,OAAzB,C;K;IAGrB,wD;MAWgD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MAC/E,oCAAa,2BA
AkB,SAAlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;
K;IAGrB,wD;MAWkD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MACjF,oCAAa,2BAAkB,SAAlB,EA
A6B,OAA7B,EAAsC,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;K;IAGrB,wD;M
AW8C,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MAC7E,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EA
AsC,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;K;IAGrB,wD;MAWgD,yB;QAA
A,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MAC/E,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;
MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;K;IAGrB,wD;MAWkD,yB;QAAA,YAAiB,C;MA
AG,uB;QAAA,UAAe,gB;MACjF,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;MACR,SAAY,M
AAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;K;IAGrB,wD;MAWoD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,U
AAe,gB;MACnF,oCAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EA
Ac,SAAd,EAAyB,OAAzB,C;K;IAGrB,yD;MAWsD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MACrF,o
CAAa,2BAAkB,SAAlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,
OAAzB,C;K;IAGrB,yD;MAWgD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MAC/E,oCAAa,2BAAkB,S
AAlB,EAA6B,OAA7B,EAAsC,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;K;iFA
GrB,8B;MAKI,OAAY,SAAY,QAAO,CAAQ,OAAR,CAAP,C;K;iFAG5B,yB;MAwIA,iD;MAxIA,qC;QAKI,OAwI
O,gCAxIK,eAAY,OAAZ,EAwIL,C;O;KA7IX,C;iFAQA,yB;MAwIA,iD;MAxIA,qC;QAKI,OAwIO,gCAxIK,gBA
Aa,OAAb,EAwIL,C;O;KA7IX,C;iFAQA,yB;MAwIA,iD;MAxIA,qC;QAKI,OAwIO,gCAxIK,gBAAW,OAAX,EA
wIL,C;O;KA7IX,C;iFAQA,yB;MAwIA,iD;MAxIA,qC;QAKI,OAwIO,gCAxIK,mBAAY,OAAZ,CAwIL,C;O;KA7
IX,C;iFAQA,yB;MAwIA,iD;MAxIA,qC;QAKI,OAwIO,gCAxIK,kBAAa,OAAb,EAwIL,C;O;KA7IX,C;gFAQA,y
B;MAwIA,iD;MAxIA,qC;QAKI,OAwIO,gCAxIK,kBAAc,OAAd,EAwIL,C;O;KA7IX,C;iFAQA,yB;MAwIA,iD;M
AxIA,qC;QAKI,OAwIO,gCAxIK,sBAAe,OAAf,CAwIL,C;O;KA7IX,C;iFAQA,yB;MAwIA,iD;MAxIA,qC;QAKI,
OAwIO,gCAxIK,mBAAY,OAAZ,CAwIL,C;O;KA7IX,C;IAQA,sC;MAKI,OAAO,oBAAoB,SAApB,EAA0B,QAA
1B,C;K;IAGX,sC;MAII,OAAO,mBAAwB,UAAL,SAAK,EAAO,mBAAO,QAAS,KAAhB,IAAP,CAAxB,EAAsD,
SAAK,OAA3D,EAAiE,QAAjE,C;K;IAGX,sC;MAII,OAAO,mBAAwB,UAAL,SAAK,EAAO,mBAAO,QAAS,KA
AhB,IAAP,CAAxB,EAAsD,SAAK,OAA3D,EAAiE,QAAjE,C;K;IAGX,sC;MAII,OAAO,mBAAwB,UAAL,SAAK
,EAAO,mBAAO,QAAS,KAAhB,IAAP,CAAxB,EAAsD,SAAK,OAA3D,EAAiE,QAAjE,C;K;IAGX,sC;MAII,OA
AO,oBAAoB,SAApB,EAA0B,QAA1B,C;K;IAGX,sC;MAII,OAAO,mBAAwB,UAAL,SAAK,EAAO,mBAAO,QA
AS,KAAhB,IAAP,CAAxB,EAAsD,SAAK,OAA3D,EAAiE,QAAjE,C;K;IAGX,sC;MAII,OAAO,mBAAwB,UAAL
,SAAK,EAAO,mBAAO,QAAS,KAAhB,IAAP,CAAxB,EAAsD,SAAK,OAA3D,EAAiE,QAAjE,C;K;IAGX,sC;M
AII,OAAO,oBAAoB,SAApB,EAA0B,QAA1B,C;K;IAGX,sC;MAII,OAAO,mBAAwB,UAAL,SAAK,EAAO,mBA
AO,QAAS,KAAhB,IAAP,CAAxB,EAAsD,SAAK,OAA3D,EAAiE,QAAjE,C;K;iFAGX,+B;MAKI,OAAY,SAAY,
QAAO,QAAP,C;K;iFAG5B,yB;MAAA,iD;MAAA,sC;QAKI,OAAO,qBAAqB,SAArB,EAA2B,QAA3B,C;O;KAL
X,C;iFAQA,yB;MAAA,iD;MAAA,sC;QAKI,OAAO,qBAAqB,SAArB,EAA2B,QAA3B,C;O;KALX,C;iFAQA,yB;
MAAA,iD;MAAA,sC;QAKI,OAAO,qBAAqB,SAArB,EAA2B,QAA3B,C;O;KALX,C;iFAQA,yB;MAAA,iD;MA
AA,sC;QAKI,OAAO,qBAAqB,SAArB,EAA2B,QAA3B,C;O;KALX,C;iFAQA,yB;MAAA,iD;MAAA,sC;QAKI,O
AAO,qBAAqB,SAArB,EAA2B,QAA3B,C;O;KALX,C;iFAQA,yB;MAAA,iD;MAAA,sC;QAKI,OAAO,qBAAqB,
SAArB,EAA2B,QAA3B,C;O;KALX,C;iFAQA,yB;MAAA,iD;MAAA,sC;QAKI,OAAO,qBAAqB,SAArB,EAA2B,
QAA3B,C;O;KALX,C;iFAQA,yB;MAAA,iD;MAAA,sC;QAKI,OAAO,qBAAqB,SAArB,EAA2B,QAA3B,C;O;K
ALX,C;8FAQA,8B;MAKI,OAAY,SAAY,QAAO,CAAQ,OAAR,CAAP,C;K;IAoBL,2B;MAAsB,OAAA,CAAE,iB

AAU,CAAV,C;K;IAP/C,2B;MAOI,IAAI,mBAAO,CAAX,C;QAwQY,eAxQO,WAwQP,C;;K;IAnNhB,2B;MAQI,IAAI,mBAAO,CAAX,C;QAAc,UAAU,SAAV,C;K;IAGlB,wC;MAQI,IAAI,mBAAO,CAAX,C;QAAc,cAAc,SAAd,EAAoB,UAApB,C;K;IAGlB,gD;MAewD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MACvF,oCAAa,2BAAkB,SAAlB,EAA6B,EAA7B,EAAiC,OAA7B,EAAsC,gBAAtC,C;MACb,gBAAc,SAAd,EAAoB,SAApB,EAA+B,EAA/B,B,OAAiC,B,EAAwC,wC,cAAxC,C;K;IAGJ,gD;MAAiC,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MAChE,oCAAa,2BAAkB,SAAlB,EAA6B,EAA7B,EAAiC,OAA7B,EAAsC,gBAAtC,C;MACb,eAAoB,SAApB,EAAgC,UAAhC,EAA4C,SAA5C,EAAuD,EAAvD,EAA2D,OAA3D,EAAoE,OAApE,B,C;MACvB,KAAT,SAAS,C;K;IAGb,gD;MAAkC,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MACjE,oCAAa,2BAAkB,SAAlB,EAA6B,EAA7B,EAAiC,OAA7B,EAAsC,gBAAtC,C;MACb,eAAoB,SAApB,EAAgC,UAAhC,EAA4C,SAA5C,EAAuD,EAAvD,EAA2D,OAA3D,EAAoE,OAApE,B,C;MACvB,KAAT,SAAS,C;K;IAGb,gD;MAAgC,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MAC/D,oCAAa,2BAAkB,SAAlB,EAA6B,EAA7B,EAAiC,OAA7B,EAAsC,gBAAtC,C;MACb,eAAoB,SAApB,EAAgC,UAAhC,EAA4C,SAA5C,EAAuD,EAAvD,EAA2D,OAA3D,EAAoE,OAApE,B,C;MACvB,KAAT,SAAS,C;K;IAGb,gD;MAAiC,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MAChE,oCAAa,2BAAkB,SAAlB,EAA6B,EAA7B,EAAiC,OAA7B,EAAsC,gBAAtC,C;MACb,gBAAc,SAAd,EAAoB,EAApB,EAAwB,C,OAAxB,EAAiC,EAAjC,EAAqC,OAArC,EAA8C,OAA9C

B,yB;MAkBA,iB;MAlBA,uB;QAMI,OAkBO,MAAO,KAlBC,CAkBD,EAlBY,CAkBZ,C;O;KAxBlB,C;mFASA,yB;MASA,iB;MATA,uB;QAMI,OASO,MAAO,KATC,CASD,EATY,CASZ,C;O;KAflB,C;mFASA,yB;MAAA,iB;MAAA,uB;QAMI,OAAO,MAAO,KAAI,CAAJ,EAAO,CAAP,C;O;KANlB,C;mFASA,gB;MAMI,OAAW,kBAAK,CAAL,MAAJ,GAAY,CAAZ,MAAZ,GAAqB,CAArB,MAAyB,C;K;mFAG9B,yB;MAAA,iB;MAAA,uB;QAQI,OAAO,MAAO,KAAI,CAAJ,EAAO,CAAP,EAAU,CAAV,C;O;KARlB,C;mFAWA,yB;MAAA,iB;MAAA,uB;QAQI,OAAO,MAAO,KAAI,CAAJ,EAAO,CAAP,EAAU,CAAV,C;O;KARlB,C;IAWA,2B;MAOI,OAAO,SAAM,CAAN,EAAS,SAAM,CAAN,EAAS,CAAT,EAAY,CAAZ,C;K;mFAGhC,yB;MAAA,iB;MAAA,0B;QAMI,OAAO,MAAO,KAAM,CAAN,EAAS,EAAT,iB,CAAa,EAAb,iB,CAAiB,EAAjB,4B,CAAqB,EAArB,4B,CAAyB,C;O;KANlB,C;mFASA,yB;MAAA,iB;MAAA,0B;QAMI,OAAO,MAAO,KAAM,CAAN,EAAS,EAAT,iB,CAAa,EAAb,iB,CAAiB,EAAjB,4B,CAAqB,EAArB,4B,CAAyB,C;O;KANlB,C;mFASA,yB;MAAA,iB;MAAA,0B;QAMI,OAAO,MAAO,KAAI,CAAJ,EAAO,CAAP,EAAU,CAAV,EAAa,CAAb,EAAgB,CAAhB,C;O;KANlB,C;mFASA,mB;MAMW,UAAe,CAAf,EAAkB,iBAAoB,CAApB,EAAuB,CAAvB,EAA0B,CAA1B,EAA6B,CAA7B,EAAgC,CAAhC,EAAmC,CAAnC,C;O;KANlB,C;mFASA,yB;MAAA,iB;MAAA,0B;QAQI,OAAO,MAAO,KAAI,CAAJ,EAAO,CAAP,EAAU,CAAV,EAAa,CAAb,EAAgB,CAAhB,EAAmB,CAAnB,EAAsB,CAAtB,C;O;KARlB,C;mFAWA,yB;MAAA,iB;MAAA,0B;QAQI,OAAO,MAAO,KAAI,CAAJ,EAAO,CAAP,EAAU,CAAV,EAAa,CAAb,EAAgB,CAAhB,EAAmB,CAAnB,EAAsB,CAAtB,C;O;KARlB,C;IAWA,4B;MAQI,OAAO,MAAO,KAAI,CAAJ,EAAO,CAAP,EAAU,CAAV,EAAa,CAAb,EAAgB,CAAhB,EAAmB,CAAnB,EAAsB,CAAtB,C;O;KARlB,C;mFAWA,yB;MAAA,iB;MAAA,0B;QAQI,OAAO,MAAO,KAAI,CAAJ,EAAO,CAAP,EAAU,CAAV,EAAa,CAAb,EAAgB,CAAhB,EAAmB,CAAnB,EAAsB,CAAtB,C;O;KARlB,C;mFAWA,yB;MAAA,iB;MAAA,0B;QAQI,OAAO,MAAO,KAAI,CAAJ,EAAO,CAAP,EAAU,CAAV,EAAa,CAAb,EAAgB,CAAhB,EAAmB,CAAnB,EAAsB,CAAtB,C;O;KARlB,C;mFAW

A,qB;MAEI,0BAA0B,gBACtB,EADsB,EACd,IADc,EACN,IADM,EACE,IADF,EACU,IADV,EACkB,IADlB,EAC0B,IAD1B,EACkC,IADlC,EAC0C,IAD1C,EACkD,IADlD,EAC0D,IAD1D,EACkE,IADlE,EAC0E,IAD1E,EACkF,IADlF,EAC0F,IAD1F,EACkG,IADlG,EAC0G,IAD1G,EACkH,IADlH,EAC0H,IAD1H,EACkI,IADlI,EAEtB,IAFsB,EAEd,IAFc,EAEN,IAFM,EAEE,IAFF,EAEU,IAFV,EAEkB,IAFlB,EAE0B,IAF1B,EAEkC,IAFlC,EAE0C,IAF1C,EAEkD,KAFlD,EAE0D,KAF1D,EAEkE,KAFlE,EAE0E,KAF1E,EAEkF,KAFlF,EAE0F,KAF1F,EAEkG,KAFlG,EAE0G,KAF1G,E;K;;;IAF9B,6B;MAAA,4B;QAAA,W;;MAAA,qB;K;IAQA,0C;MAKI,aAAa,C;MACb,UAAAU,KAAM,OAAN,GAAa,CAAb,I;MACV,aAAa,E;MACb,YAAY,C;MACZ,OAAO,UAAU,GAAjB,B,C;QACI,SAAS,CAAC,SAAS,GAAT,IAAD,IAAiB,B,CAAjB,I;QACT,QAAQ,MAAM,MAAN,C;QACR,IAAI,SAAS,KAAb,B,C;UACI,QAAQ,OAAO,M;;UAEP,MAAM,SAAS,CAAT,I;;MAEd,OAAO,UAAc,c,SAAS,KAAT,B,GAAiB,C,GAAjB,C,C;MAClB,KAAK,C;;IAGT,WAAW,C;MAGP,SAAS,S3CCiC,I;M2CA1C,YAAY,kBAAkB,mBAAM,mBAAxB,EAAoC,EAApC,C;MACZ,WAAW,KAAK,mBAAM,mBAAN,CAAiB,KAAjB,B,CAAuB,KAAvB,I;MACX,OAAO,OAAO,EAAX,IAAe,GAAf,E;MACA

,iCAAuB,cAAvB,MAA1B,C;;MAAtC,a;K;IAGJ,wC;MxCuTe,WwC/SY,KxC+SZ,IAAS,C;MAAT,S;QAAc,OwC/SF,KxC+SE,IA6/GT,gBAAR,iBAAQ,C;;MA7/GT,U;MAAA,S;QAAA,SAAsC,sBwC/StB,KxC+SsB,C;;QwC/Sb,MAAM,8BAA0B,iCAAuB,cAAvB,MAA1B,C;;MAAtC,a;K;IAGJ,wC;MxCwTe,WwChTY,KxCgTZ,IAAS,C;MAAT,S;QAAc,OwChTF,KxCgTE,IAy/GT,gBAAR,iBAAQ,C;;MAz/GT,U;MAAA,S;QAAA,SAAsC,sBwChTtB,KxCgTsB,C;;QwChTb,MAAM,8BAA0B,iCAAuB,cAAvB,MAA1B,C;;MAAtC,a;K;IASO,6C;MAAA,8B;MAAS,uB;K;8FACW,Y;MAAQ,OAAA,gBAAY,K;K;+CAC3C,Y;MAAkC,OAAA,gBAAY,U;K;sDAC9C,mB;MAAgD,OAAA,gBAAY,gBAAS,OAAT,C;K;mDAC5D,iB;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,SAAzB,C;MACb,OAAO,6BAAY,KAAZ,EAAmB,C;K;qDAEX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCsrBO,UwCtrBA,gBxCsrBR,QAAQ,EwCtrBC,oBO3EgOF,KmCsdlB,C;K;yDwCtrBX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCy6BO,cwCz6BA,gBxCy6BR,QAAQ,EwCz6BC,oBO3E2E2NN,KmC8sBlB,C;K;;IwC/7Bb,mB;MAAS,uB;K;8FACW,Y;MAAQ,OAAA,gBAAY,K;K;+CAC3C,Y;MAAkC,OAAA,gBAAY,U;K;sDAC9C,mB;MAAiD,OAAA,gBAAY,gBAAS,OAAT,C;K;mDAC7D,iB;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,SAAzB,C;MACb,OAAO,6BAAY,KAAZ,EAAmB,C;K;qDAEX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCqqBO,UwCrqBA,gBxCqqBR,QAAQ,EwCrqBC,oBO3EgOF,KmCsdlB,C;K;yDwCrqBX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCw5BO,cwCx5BA,gBxCw5BR,QAAQ,EwCx5BC,oBO3E2E2NN,KmC8sBlB,C;K;;IwC/6Bb,mB;MAAS,uB;K;8FACW,Y;MAAQ,OAAA,gBAAY,K;K;+CAC3C,Y;MAAkC,OAAA,gBAAY,U;K;sDAC9C,mB;MAAiD,OAAA,gBAAY,gBAAS,OAAT,C;K;mDAC7D,iB;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,SAAzB,C;MACb,OAAO,6BAAY,KAAZ,EAAmB,C;K;qDAEX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCo4BO,UwCp4BA,gBxCo4BR,QAAQ,EwCp4BC,oBO3EgOF,KmCsdlB,C;K;yDwCp4BX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCu3BO,cwCv3BA,gBxCu3BR,QAAQ,EwCv3BC,oBO3E2E2NN,KmC8sBlB,C;K;;IwC/4Bb,mB;MAAS,uB;K;8FACW,Y;MAAQ,OAAA,gBAAY,K;K;+CAC3C,Y;MAAkC,OAAA,gBAAY,U;K;sDAC9C,mB;MAAiD,OAAA,gBAAY,gBAAS,OAAT,C;K;mDAC7D,iB;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,SAAzB,C;MACb,OAAO,6BAAY,KAAZ,EAAmB,C;K;qDAEX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCm2BO,UwCn2BA,gBxCm2BR,QAAQ,EwCn2BC,oBO3EgOF,KmCsdlB,C;K;yDwCn2BX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCs1BO,cwCt1BA,gBxCs1BR,QAAQ,EwCt1BC,oBO3E2E2NN,KmC8sBlB,C;K;;IwC95Bb,mB;MAAS,uB;K;8FACW,Y;MAAQ,OAAA,gBAAY,K;K;+CAC3C,Y;MAAkC,OAAA,gBAAY,U;K;sDAC9C,mB;MAAiD,OAAA,gBAAY,gBAAS,OAAT,C;K;mDAC7D,iB;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,SAAzB,C;MACb,OAAO,6BAAY,KAAZ,EAAmB,C;K;qDAEX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCk0BO,UwCl0BA,gBxCk0BR,QAAQ,EwCl0BC,oBO3EgOF,KmCsdlB,C;K;yDwCl0BX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCqzBO,cwCrzBA,gBxCqzBR,QAAQ,EwCrzBC,oBO3E2E2NN,KmC8sBlB,C;K;;IwC75Bb,mB;MAAS,uB;K;8FACW,Y;MAAQ,OAAA,gBAAY,K;K;+CAC3C,Y;MAAkC,OAAA,gBAAY,U;K;sDAC9C,mB;MAAiD,OAAA,gBAAY,gBAAS,OAAT,C;K;mDAC7D,iB;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,SAAzB,C;MACb,OAAO,6BAAY,KAAZ,EAAmB,C;K;qDAEX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCi0BO,UwCj0BA,gBxCi0BR,QAAQ,EwCj0BC,oBO3EgOF,KmCsdlB,C;K;yDwCj0BX,mB;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QAAgC,OAAO,E;MACvC,OxCozBO,cwCpzBA,gBxCozBR,QAAQ,EwCp

MAMuD,sBAAQ,4BAAR,C;K;IAEvD,6B;MAIwE,kBAAhB,0B;MAAwB,uB;MAAxB,OJjE7C,W;K;IImEX,4B;MAQI,gBAAgB,SAAhB,EAAsB,cAAtB,C;K;IAGJ,2C;MAQI,gBAAgB,SAAhB,EAAsB,UAAtB,C;K;IAGJ,2C;MACI,IAAI,IAAK,KAAL,IAAa,CAAjB,C;QAAoB,M;MAEpB,YAAY,YAAY,IAAZ,C;MACZ,gBAAc,KAAd,EAAqBA,UAAArB,C;MAEA,aAAU,CAAV,MAAkB,KAAM,OAAxB,M;QACI,iBAAK,CAAL,EAAU,MAAM,CAAN,CAAV,C;;K;IAIR,uC;MACI,OAAO,gBAAkB,IAAlB,O;K;IAGX,iF;MAII,oCAAa,2BAAkB,UAAlB,EAA8B,EAA9B,QAA9B,EAAAwC,MAAAO,OA/C/C/C

,0B;MACI,oCAAa,4BAAmB,KAAnB,EAA0B,YAA1B,C;MAEb,WAAK,aAAI,mBAAY,KAAZ,IAAJ,EAAuB,OAAvB,C;MACL,mC;K;wDAGJ,iB;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,YAAzB,C;MAEb,OAAO,wBAAK,mBAAY,KAAZ,IAAL,C;K;6DAGX,iB;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,YAAzB,C;MAEb,aAAa,WAAK,kBAAS,mBAAY,KAAZ,IAAT,C;MACIB,mC;MACA,OAAO,M;K;wDAGX,0B;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,YAAzB,C;MAEb,OAAO,WAAK,aAAI,mBAAY,KAAZ,IAAJ,EAAuB,OAAvB,C;K;mGAGO,Y;MAAQ,mB;K;2DAE/B,Y;MAA+C,WAAK,iB;K;;ICxMN,8B;MAAiC,oC;MAAwB,uBAAoC,I;MAA+C,yBAA6C,I;K;IAlFR,oD;MAAiC,wB;MAAwB,gBAAqB,K;K;iFAHha,Y;MAAQ,aAAa,yB;K;uGAKrB,Y;MAAQ,oB;K;8DAE9B,oB;MAAK,eAAAe,eAAA,IAAAK,S;MACpB,gBAAAC,Q;MACd,OAAO,Q;K;wDAGX,Y;MAAwC,WAAK,iBAAS,iBAAS,C;K;wDAC/D,B,Y;MAAwC,WAAK,iBAAS,iBAAS,C;K;sDAClC,iB;MAAA4C,+DAAY,IAAAZ,Z,EAAAa,kB,KAAAC,lB,C;K;;IAlB5C,8E;MAAA,aAAA,wE;MAAA,sC,2CAAK,KAAL,EAAK,KAAL,EAAK,KAAL,E

C/F,EADoH,CACpH,EADuI,gBACvI,C;MwErwBI,IAAI,KAAM,OAAN,GAAa,SAAjB,C;QACI,MAAM,SAAN,IAAc,6E;;MAGlB,OAAO,K;K;kCAGX,Y;MACI,OAAO,EAAS,MAAM,MAAK,oBAAL,C;K;yCAI1B,Y;MACI,IAAI,yBAAJ,C;QAAgB,MAAM,oC;K;+CAG1B,iB;MACI,oCAAa,kCAAyB,SAAzB,C;MADoB,Y;K;wDAIrC,iB;MACI,oCAAa,mCAA0B,SAA1B,C;MAD6B,Y;K;;IAlJ9C,+B;MAAA,mD;MAG8B,sB9BRa,E8BQb,C;MAH9B,Y;K;IAKA,kD;MAAA,mD;MAIkD,sB9BdP,E8BcO,C;MAJlD,Y;K;IAMA,2C;MAAA,mD;MAGqD,sBlENa,YkEMR,QlENQ,CkEMb,C;MAHrD,Y;K;ICrBJ,0C;MACI,IAAI,6BAAJ,C;QACU,KAAY,MAAK,UAAL,C;;QAElB,UAAU,KAAV,EAAwC,CAAxC,EAAiD,cAAN,KAAM,CAAjD,EAA4D,eAAW,UAAX,CAA5D,EAAmF,CAAnF,C;;K;IAMiB,kD;MAAA,uB;QAAA,OAAK,KAAL,EAAY,kBAAZ,EAAwB,KAAxB,EAAgC,KAAhC,EAAwC,KAAxC,EAAgD,KAAhD,EAAwD,KAAxD,EAAgE,KAAhE,EAAwE,KAAxE,EAAgF,KAAhF,EAAwF,KAAxF,EAAgG,KAAhG,EAAwG,KAAxG,EAAgH,KAAhH,EAAwH,KAAxH,EAAgI,KAAhI,EAAwI,KAAxI,EAAgJ,KAAhJ,EAAwJ,KAAxJ,EAAgK,KAAhK,EAAwK,KAAxK,EAAgL,KAAhL,EAAwL,KAAxL,EAAgM,KAAhM,EAAwM,KAAxM,EAAgN,KAAhN,EAAwN,KAAxN,EAAgO,KAAhO,EAAwO,KAAxO,EAAgP,KAAhP,EAAwP,KAAxP,EAAgQ,KAAhQ,EAAwQ,KAAxQ,EAAgR,KAAhR,EAAwR,KAAxR,EAAgS,KAAhS,EAAwS,KAAxS,EAAgT,KAAhT,EAAwT,KAAxT,EAAgU,KAAhU,EAAwU,KAAxU,EAAgV,KAAhV,EAAwV,KAAxV,EAAgW,KAAhW,EAAwW,KAAxW,EAAgX,KAAhX,EAAwX,KAAxX,EAAgY,KAAhY,EAAwY,KAAxY,EAAgZ,KAAhZ,EAAwZ,KAAxZ,EAAga,KAAha,EAAwa,KAAxa,EAAgb,KAAhb,EAAwb,KAAxb,EAAgc,KAAhc,EAAwc,KAAxc,EAAgd,KAAhd,EAAwd,KAAxd,EAAge,KAAhe,EAAwe,KAAxe,EAAgf,KAAhf,EAAwf,KAAxf,EAAgg,KAAhg,EAAwg,KAAxg,EAAgh,KAAhh,EAAwh,KAAxh,EAAgi,KAAhi,EAAwi,KAAxi,EAAgj,KAAhj,EAAwj,KAAxj,EAAgk,KAAhk,EAAwk,KAAxk,EAAgl,KAAhl,EAAwl,KAAxl,EAAgm,KAAhm,EAAwm,KAAxm,EAAgn,KAAhn,EAAwn,KAAxn,EAAgo,KAAho,EAAwo,KAAxo,EAAgp,KAAhp,EAAwp,KAAxp,EAAgq,KAAhq,EAAwq,KAAxq,EAAgr,KAAhr,EAAwr,KAAxr,EAAgs,KAAhs,EAAwsB,KAAxsB,EAAgtB,KAAhtB,EAAwtB,KAAxtB,EAAguB,KAAhuB,EAAwuB,KAAxuB,EAAgvB,KAAhvB,EAAwvB,KAAxvB,EAAgwB,KAAhwB,EAAwwB,KAAxwB,EAAgxB,KAAhxB,EAAwxB,KAAxxB,EAAgyB,KAAhyB,EAAwyB,KAAxyB,EAAgzB,KAAhzB,EAAwzB,KAAxzB,EAAg0B,KAAh0B,EAAw0B,KAAx0B,EAAg1B,KAAh1B,EAAw1B,KAAx1B,EAAg2B,KAAh2B,EAAw2B,KAAx2B,EAAg3B,KAAh3B,EAAw3B,KAAx3B,EAAg4B,KAAh4B,EAAw4B,KAAx4B,EAAg5B,KAAh5B,EAAw5B,KAAx5B,EAAg6B,KAAh6B,EAAw6B,KAAx6B,EAAg7B,KAAh7B,EAAw7B,KAAx7B,EAAg8B,KAAh8B,EAAw8B,KAAx8B,EAAg9B,KAAh9B,EAAw9B,KAAx9B,EAAg+B,KAAh+B,EAAw+B,KAAx+B,EAAg/B,KAAh/B,EAAw/B,KAAx/B,EAAggC,KAAhgC,EAAwgC,KAAxgC,EAAghC,KAAhhC,EAAwhC,KAAxhC,EAAgiC,KAAhiC,EAAwiC,KAAxiC,EAAgjC,KAAhjC,EAAwjC,KAAxjC,EAAgkC,KAAhkC,EAAwkC,KAAxkC,EAAglC,KAAhlC,EAAwlC,KAAxlC,EAAgmC,KAAhmC,EAAwmC,KAAxmC,EAAgnC,KAAhnC,EAAwnC,KAAxnC,EAAgoC,KAAhoC,EAAwoC,KAAxoC,EAAgpC,KAAhpC,EAAwpC,KAAxpC,EAAgqC,KAAhqC,EAAwqC,KAAxqC,EAAgrC,KAAhrC,EAAwrC,KAAxrC,EAAgsC,KAAhsC,EAAwsC,KAAxsC,EAAgtC,KAAhtC,EAAwtC,KAAxtC,EAAguC,KAAhuC,EAAwuC,KAAxuC,EAAgvC,KAAhvC,EAAwvC,KAAxvC,EAAgwC,KAAhwC,EAAwwC,KAAxwC,EAAgxC,KAAhxC,EAAwxC,KAAxxC,EAAgyC,KAAhyC,EAAwyC,KAAxyC,EAAgzC,KAAhzC,EAAwzC,KAAxzC,EAAg0C,KAAh0C,EAAw0C,KAAx0C,EAAg1C,KAAh1C,EAAw1C,KAAx1C,EAAg2C,KAAh2C,EAAw2C,KAAx2C,EAAg3C,KAAh3C,EAAw3C,KAAx3C,EAAg4C,KAAh4C,EAAw4C,KAAx4C,EAAg5C,KAAh5C,EAAw5C,KAAx5C,EAAg6C,KAAh6C,EAAw6C,KAAx6C,EAAg7C,KAAh7C,EAAw7C,KAAx7C,EAAg8C,KAAh8C,EAAw8C,KAAx8C,EAAg9C,KAAh9C,EAAw9C,KAAx9C,EAAg+C,KAAh+C,EAAw+C,KAAx+C,EAAg/C,KAAh/C,EAAw/C,KAAx/C,EAAggD,KAAhgD,EAAwgD,KAAxgD,EAAghD,KAAhhD,EAAwhD,KAAxhD,EAAgiD,KAAhiD,EAAwiD,KAAxiD,EAAgjD,KAAhjD,EAAwjD,KAAxjD,EAAgkD,KAAhkD,EAAwkD,KAAxkD,EAAglD,KAAhlD,EAAwlD,KAAxlD,EAAgmD,KAAhmD,EAAwmD,KAAxmD,EAAgnD,KAAhnD,EAAwnD,KAAxnD,EAAgoD,KAAhoD,EAAwoD,KAAxoD,EAAgpD,KAAhpD,EAAwpD,KAAxpD,EAAgqD,KAAhqD,EAAwqD,KAAxqD,EAAgrD,KAAhrD,EAAwrD,KAAxrD,EAAgsD,KAAhsD,EAAwsD,KAAxsD,EAAgtD,KAAhtD,EAAwtD,KAAxtD,EAAguD,KAAhuD,EAAwuD,KAAxuD,EAAgvD,KAAhvD,EAAwvD,KAAxvD,EAAgwD,KAAhwD,EAAwwD,KAAxwD,EAAgxD,KAAhxD,EAAwxD,KAAxxD,EAAgyD,KAAhyD,EAAwyD,KAAxyD,EAAgzD,KAAhzD,EAAwzD,KAAxzD,EAAg0D,KAAh0D,EAAw0D,KAAx0D,EAAg1D,KAAh1D,EAAw1D,KAAx1D,EAAg2D,KAAh2D,EAAw2D,KAAx2D,EAAg3D,KAAh3D,EAAw3D,KAAx3D,EAAg4D,KAAh4D,EAAw4D,KAAx4D,EAAg5D,KAAh5D,EAAw5D,KAAx5D,EAAg6D,KAAh6D,EAAw6D,KAAx6D,EAAg7D,KAAh7D,EAAw7D,KAAx7D,EAAg8D,KAAh8D,EAAw8D,KAAx8D,EAAg9D,KAAh9D,EAAw9D,KAAx9D,EAAg+D,KAAh+D,EAAw+D,KAAx+D,EAAg/D,KAAh/D,EAAw/D,KAAx/D,EAAggE,KAAhgE,EAAwgE,KAAxgE,EAAghE,KAAhhE,EAAwhE,KAAxhE,EAAgiE,KAAhiE,EAAwiE,KAAxiE,EAAgjE,KAAhjE,EAAwjE,KAAxjE,EAAgkE,KAAhkE,EAAwkE,KAAxkE,EAAglE,KAAhlE,EAAwlE,KAAxlE,EAAgmE,KAAhmE,EAAwmE,KAAxmE,EAAgnE,KAAhnE,EAAwnE,KAAxnE,EAAgoE,KAAhoE,EAAwoE,KAAxoE,EAAgpE,KAAhpE,EAAwpE,KAAxpE,EAAgqE,KAAhqE,EAAwqE,KAAxqE,EAAgrE,KAAhrE,EAAwrE,KAAxrE,EAAgsE,KAAhsE,EAAwsE,KAAxsE,EAAgtE,KAAhtE,EAAwtE,KAAxtE,EAAguE,KAAhuE,EAAwuE,KAAxuE,EAAgvE,KAAhvE,EAAwvE,KAAxvE,EAAgwE,KAAhwE,EAAwwE,KAAxwE,EAAgxE,KAAhxE,EAAwxE,KAAxxE,EAAgyE,KAAhyE,EAAwyE,KAAxyE,EAAgzE,KAAhzE,EAAwzE,KAAxzE,EAAg0E,KAAh0E,EAAw0E,KAAx0E,EAAg1E,KAAh1E,EAAw1E,KAAx1E,EAAg2E,KAAh2E,EAAw2E,KAAx2E,EAAg3E,KAAh3E,EAAw3E,KAAx3E,EAAg4E,KAAh4E,EAAw4E,KAAx4E,EAAg5E,KAAh5E,EAAw5E,KAAx5E,EAAg6E,KAAh6E,EAAw6E,KAAx6E,EAAg7E,KAAh7E,EAAw7E,KAAx7E,EAAg8E,KAAh8E,EAAw8E,KAAx8E,EAAg9E,KAAh9E,EAAw9E,KAAx9E,EAAg+E,KAAh+E,EAAw+E,KAAx+E,EAAg/E,KAAh/E,EAAw/E,KAAx/E,EAAggF,KAAhgF,EAAwgF,KAAxgF,EAAghF,KAAhhF,EAAwhF,KAAxhF,EAAgiF,KAAhiF,EAAwiF,KAAxiF,EAAgjF,KAAhjF,EAAwjF,KAAxjF,EAAgkF,KAAhkF,EAAwkF,KAAxkF,EAAglF,KAAhlF,EAAwlF,KAAxlF,EAAgmF,KAAhmF,EAAwmF,KAAxmF,EAAgnF,KAAhnF,EAAwnF,KAAxnF,EAAgoF,KAAhoF,EAAwoF,KAAxoF,EAAgpF,KAAhpF,EAAwpF,KAAxpF,EAAgqF,KAAhqF,EAAwqF,KAAxqF,EAAgrF,KAAhrF,EAAwrF,KAAxrF,EAAgsF,KAAhsF,EAAwsF,KAAxsF,EAAgtF,KAAhtF,EAAwtF,KAAxtF,EAAguF,KAAhuF,EAAwuF,KAAxuF,EAAgvF,KAAhvF,EAAwvF,KAAxvF,EAAgwF,KAAhwF,EAAwwF,KAAxwF,EAAgxF,KAAhxF,EAAwxF,KAAxxF,EAAgyF,KAAhyF,EAAwyF,KAAxyF,EAAgzF,KAAhzF,EAAwzF,KAAxzF,EAAg0F,KAAh0F,EAAw0F,KAAx0F,EAAg1F,KAAh1F,EAAw1F,KAAx1F,EAAg2F,KAAh2F,EAAw2F,KAAx2F,EAAg3F,KAAh3F,EAAw3F,KAAx3F,EAAg4F,KAAh4F,EAAw4F,KAAx4F,EAAg5F,KAAh5F,EAAw5F,KAAx5F,EAAg6F,KAAh6F,EAAw6F,KAAx6F,EAAg7F,KAAh7F,EAAw7F,KAAx7F,EAAg8F,KAAh8F,EAAw8F,KAAx8F,EAAg9F,KAAh9F,EAAw9F,KAAx9F,EAAg+F,KAAh+F,EAAw+F,KAAx+F,EAAg/F,KAAh/F,EAAw/F,KAAx/F,EAAggG,KAAhgG,EAAwgG,KAAxgG,EAAghG,KAAhhG,EAAwhG,KAAxhG,EAAgiG,KAAhiG,EAAwiG,KAAxiG,EAAgjG,KAAhjG,EAAwjG,KAAxjG,EAAgkG,KAAhkG,EAAwkG,KAAxkG,EAAglG,KAAhlG,EAAwlG,KAAxlG,EAAgmG,KAAhmG,EAAwmG,KAAxmG,EAAgnG,KAAhnG,EAAwnG,KAAxnG,EAAgoG,KAAhoG,EAAwoG,KAAxoG,EAAgpG,KAAhpG,EAAwpG,KAAxpG,EAAgqG,KAAhqG,EAAwqG,KAAxqG,EAAgrG,KAAhrG,EAAwrG,KAAxrG,EAAgsG,KAAhsG,EAAwsG,KAAxsG,EAAgtG,KAAhtG,EAAwtG,KAAxtG,EAAguG,KAAhuG,EAAwuG,KAAxuG,EAAgvG,KAAhvG,EAAwvG,KAAxvG,EAAgwG,KAAhwG,EAAwwG,KAAxwG,EAAgxG,KAAhxG,EAAwxG,KAAxxG,EAAgyG,KAAhyG,EAAwyG,KAAxyG,EAAgzG,KAAhzG,EAAwzG,KAAxzG,EAAg0G,KAAh0G,EAAw0G,KAAx0G,EAAg1G,KAAh1G,EAAw1G,KAAx1G,EAAg2G,KAAh2G,EAAw2G,KAAx2G,EAAg3G,KAAh3G,EAAw3G,KAAx3G,EAAg4G,KAAh4G,EAAw4G,KAAx4G,EAAg5G,KAAh5G,EAAw5G,KAAx5G,EAAg6G,KAAh6G,EAAw6G,KAAx6G,EAAg7G,KAAh7G,EAAw7G,KAAx7G,EAAg8G,KAAh8G,EAAw8G,KAAx8G,EAAg9G,KAAh9G,EAAw9G,KAAx9G,EAAg+G,KAAh+G,EAAw+G,KAAx+G,EAAg/G,KAAh/G,EAAw/G,KAAx/G,EAAggH,KAAhgH,EAAwgH,KAAxgH,EAAghH,KAAhhH,EAAwhH,KAAxhH,EAAgiH,KAAhiH,EAAwiH,KAAxiH,EAAgjH,KAAhjH,EAAwjH,KAAxjH,EAAgkH,KAAhkH,EAAwkH,KAAxkH,EAAglH,KAAhlH,EAAwlH,KAAxlH,EAAgmH,KAAhmH,EAAwmH,KAAxmH,EAAgnH,KAAhnH,EAAwnH,KAAxnH,EAAgoH,KAAhoH,EAAwoH,KAAxoH,EAAgpH,KAAhpH,EAAwpH,KAAxpH,EAAgqH,KAAhqH,EAAwqH,KAAxqH,EAAgrH,KAAhrH,EAAwrH,KAAxrH,EAAgsH,KAAhsH,EAAwsH,KAAxsH,EAAgtH,KAAhtH,EAAwtH,KAAxtH,EAAguH,KAAhuH,EAAwuH,KAAxuH,EAAgvH,KAAhvH,EAAwvH,KAAxvH,EAAgwH,KAAhwH,EAAwwH,KAAxwH,EAAgxH,KAAhxH,EAAwxH,KAAxxH,EAAgyH,KAAhyH,EAAwyH,KAAxyH,EAAgzH,KAAhzH,EAAwzH,KAAxzH,EAAg0H,KAAh0H,EAAw0H,KAAx0H,EAAg1H,KAAh1H,EAAw1H,KAAx1H,EAAg2H,KAAh2H,EAAw2H,KAAx2H,EAAg3H,KAAh3H,EAAw3H,KAAx3H,EAAg4H,KAAh4H,EAAw4H,KAAx4H,EAAg5H,KAAh5H,EAAw5H,KAAx5H,EAAg6H,KAAh6H,EAAw6H,KAAx6H,EAAg7H,KAAh7H,EAAw7H,KAAx7H,EAAg8H,KAAh8H,EAAw8H,KAAx8H,EAAg9H,KAAh9H,EAAw9H,KAAx9H,EAAg+H,KAAh+H,EAAw+H,KAAx+H,EAAg/H,KAAh/H,EAAw/H,KAAx/H,EAAggI,KAAhgI,EAAwgI,KAAxgI,EAAghI,KAAhhI,EAAwhI,KAAxhI,EAAgiI,KAAhiI,EAAwiI,KAAxiI,EAAgjI,KAAhjI,EAAwjI,KAAxjI,EAAgkI,KAAhkI,EAAwkI,KAAxkI,EAAglI,KAAhlI,EAAwlI,KAAxlI,EAAgmI,KAAhmI,EAAwmI,KAAxmI,EAAgnI,KAAhnI,EAAwnI,KAAxnI,EAAgoI,KAAhoI,EAAwoI,KAAxoI,EAAgpI,KAAhpI,EAAwpI,KAAxpI,EAAgqI,KAAhqI,EAAwqI,KAAxqI,EAAgrI,KAAhrI,EAAwrI,KAAxrI,EAAgsI,KAAhsI,EAAwsI,KAAxsI,EAAgtI,KAAhtI,EAAwtI,KAAxtI,EAAguI,KAAhuI,EAAwuI,KAAxuI,EAAgvI,KAAhvI,EAAwvI,KAAxvI,EAAgwI,KAAhwI,EAAwwI,KAAxwI,EAAgxI,KAAhxI,EAAwxI,KAAxxI,EAAgyI,KAAhyI,EAAwyI,KAAxyI,EAAgzI,KAAhzI,EAAwzI,KAAxzI,EAAg0I,KAAh0I,EAAw0I,KAAx0I,EAAg1I,KAAh1I,EAAw1I,KAAx1I,EAAg2I,KAAh2I,EAAw2I,KAAx2I,EAAg3I,KAAh3I,EAAw3I,KAAx3I,EAAg4I,KAAh4I,EAAw4I,KAAx4I,EAAg5I,KAAh5I,EAAw5I,KAAx5I,EAAg6I,KAAh6I,EAAw6I,KAAx6I,EAAg7I,KAAh7I,EAAw7I,KAAx7I,EAAg8I,KAAh8I,EAAw8I,KAAx8I,EAAg9I,KAAh9I,EAAw9I,KAAx9I,EAAg+I,KAAh+I,EAAw+I,KAAx+I,EAAg/I,KAAh/I,EAAw/I,KAAx/I,EAAggJ,KAAhgJ,EAAwgJ,KAAxgJ,EAAghJ,KAAhhJ,EAAwhJ,KAAxhJ,EAAgiJ,KAAhiJ,EAAwiJ,KAAxiJ,EAAgjJ,KAAhjJ,EAAwjJ,KAAxjJ,EAAgkJ,KAAhkJ,EAAwkJ,KAAxkJ,EAAglJ,KAAhlJ,EAAwlJ,KAAxlJ,EAAgmJ,KAAhmJ,EAAwmJ,KAAxmJ,EAAgnJ,KAAhnJ,EAAwnJ,KAAxnJ,EAAgoJ,KAAhoJ,EAAwoJ,KAAxoJ,EAAgpJ,KAAhpJ,EAAwpJ,KAAxpJ,EAAgqJ,KAAhqJ,EAAwqJ,KAAxqJ,EAAgrJ,KAAhrJ,EAAwrJ,KAAxrJ,EAAgsJ,KAAhsJ,EAAwsJ,KAAxsJ,EAAgtJ,KAAhtJ,EAAwtJ,KAAxtJ,EAAguJ,KAAhuJ,EAAwuJ,KAAxuJ,EAAgvJ,KAAhvJ,EAAwvJ,KAAxvJ,EAAgwJ,KAAhwJ,EAAwwJ,KAAxwJ,EAAgxJ,KAAhxJ,EAAwxJ,KAAxxJ,EAAgyJ,KAAhyJ,EAAwyJ,KAAxyJ,EAAgzJ,KAAhzJ,EAAwzJ,KAAxzJ,EAAg0J,KAAh0J,EAAw0J,KAAx0J,EAAg1J,KAAh1J,EAAw1J,KAAx1J,EAAg2J,KAAh2J,EAAw2J,KAAx2J,EAAg3J,KAAh3J,EAAw3J,KAAx3J,EAAg4J,KAAh4J,EAAw4J,KAAx4J,EAAg5J,KAAh5J,EAAw5J,KAAx5J,EAAg6J,KAAh6J,EAAw6J,KAAx6J,EAAg7J,KAAh7J,EAAw7J,KAAx7J,EAAg8J,KAAh8J,EAAw8J,KAAx8J,EAAg9J,KAAh9J,EAAw9J,KAAx9J,EAAg+J,KAAh+J,EAAw+J,KAAx+J,EAAg/J,KAAh/J,EAAw/J,KAAx/J,EAAggK,KAAhgK,EAAwgK,KAAxgK,EAAghK,KAAhhK,EAAwhK,KAAxhK,EAAgiK,KAAhiK,EAAwiK,KAAxiK,EAAgjK,KAAhjK,EAAwjK,KAAxjK,EAAgkK,KAAhkK,EAAwkK,KAAxkK,EAAglK,KAAhlK,EAAwlK,KAAxlK,EAAgmK,KAAhmK,EAAwmK,KAAxmK,EAAgnK,KAAhnK,EAAwnK,KAAxnK,EAAgoK,KAAhoK,EAAwoK,KAAxoK,EAAgpK,KAAhpK,EAAwpK,KAAxpK,EAAgqK,KAAhqK,EAAwqK,KAAxqK,EAAgrK,KAAhrK,EAAwrK,KAAxrK,EAAgsK,KAAhsK,EAAwsK,KAAxsK,EAAgtK,KAAhtK,EAAwtK,KAAxtK,EAAguK,KAAhuK,EAAwuK,KAAxuK,EAAgvK,KAAhvK,EAAwvK,KAAxvK,EAAgwK,KAAhwK,EAAwwK,KAAxwK,EAAgxK,KAAhxK,EAAwxK,KAAxxK,EAAgyK,KAAhyK,EAAwyK,KAAxyK,EAAgzK,KAAhzK,EAAwzK,KAAxzK,EAAg0K,KAAh0K,EAAw0K,KAAx0K,EAAg1K,KAAh1K,EAAw1K,KAAx1K,EAAg2K,KAAh2K,EAAw2K,KAAx2K,EAAg3K,KAAh3K,EAAw3K,KAAx3K,EAAg4K,KAAh4K,EAAw4K,KAAx4K,EAAg5K,KAAh5K,EAAw5K,KAAx5K,EAAg6K,KAAh6K,EAAw6K,KAAx6K,EAAg7K,KAAh7K,EAAw7K,KAAx7K,EAAg8K,KAAh8K,EAAw8K,KAAx8K,EAAg9K,KAAh9K,EAAw9K,KAAx9K,EAAg+K,KAAh+K,EAAw+K,KAAx+K,EAAg/K,KAAh/K,EAAw/K,KAAx/K,EAAggL,KAAhgL,EAAwgL,KAAxgL,EAAghL,KAAhhL,EAAwhL,KAAxhL,EAAgiL,KAAhiL,EAAwiL,KAAxiL,EAAgjL,KAAhjL,EAAwjL,KAAxjL,EAAgkL,KAAhkL,EAAwkL,KAAxkL,EAAglL,KAAhlL,EAAwlL,KAAxlL,EAAgmL,KAAhmL,EAAwmL,KAAxmL,EAAgnL,KAAhnL,EAAwnL,KAAxnL,EAAgoL,KAAhoL,EAAwoL,KAAxoL,EAAgpL,KAAhpL,EAAwpL,KAAxpL,EAAgqL,KAAhqL,EAAwqL,KAAxqL,EAAgrL,KAAhrL,EAAwrL,KAAxrL,EAAgsL,KAAhsL,EAAwsL,KAAxsL,EAAgtL,KAAhtL,EAAwtL,KAAxtL,EAAguL,KAAhuL,EAAwuL,KAAxuL,EAAgvL,KAAhvL,EAAwvL,KAAxvL,EAAgwL,KAAhwL,EAAwwL,KAAxwL,EAAgxL,KAAhxL,EAAwxL,KAAxxL,EAAgyL,KAAhyL,EAAwyL,KAAxyL,EAAgzL,KAAhzL,EAAwzL,KAAxzL,EAAg0L,KAAh0L,EAAw0L,KAAx0L,EAAg1L,KAAh1L,EAAw1L,KAAx1L,EAAg2L,KAAh2L,EAAw2L,KAAx2L,EAAg3L,KAAh3L,EAAw3L,KAAx3L,EAAg4L,KAAh4L,EAAw4L,KAAx4L,EAAg5L,KAAh5L,EAAw5L,KAAx5L,EAAg6L,KAAh6L,EAAw6L,KAAx6L,EAAg7L,KAAh7L,EAAw7L,KAAx7L,EAAg8L,KAAh8L,EAAw8L,KAAx8L,EAAg9L,KAAh9L,EAAw9L,KAAx9L,EAAg+L,KAAh+L,EAAw+L,KAAx+L,EAAg/L,KAAh/L,EAAw/L,KAAx/L,EAAggM,KAAhgM,EAAwgM,KAAxgM,EAAghM,KAAhhM,EAAwhM,KAAxhM,EAAgiM,KAAhiM,EAAwiM,KAAxiM,EAAgjM,KAAhjM,EAAwjM,KAAxjM,EAAgkM,KAAhkM,EAAwkM,KAAxkM,EAAglM,KAAhlM,EAAwlM,KAAxlM,EAAgmM,KAAhmM,EAAwmM,KAAxmM,EAAgnM,KAAhnM,EAAwnM,KAAxnM,EAAgoM,KAAhoM,EAAwoM,KAAxoM,EAAgpM,KAAhpM,EAAwpM,KAAxpM,EAAgqM,KAAhqM,EAAwqM,KAAxqM,EAAgrM,KAAhrM,EAAwrM,KAAxrM,EAAgsM,KAAhsM,EAAwsM,KAAxsM,EAAgtM,KAAhtM,EAAwtM,KAAxtM,EAAguM,KAAhuM,EAAwuM,KAAxuM,EAAgvM,KAAhvM,EAAwvM,KAAxvM,EAAgwM,KAAhwM,EAAwwM,KAAxwM,EAAgxM,KAAhxM,EAAwxM,KAAxxM,EAAgyM,KAAhyM,EAAwyM,KAAxyM,EAAgzM,KAAhzM,EAAwzM,KAAxzM,EAAg0M,KAAh0M,EAAw0M,KAAx0M,EAAg1M,KAAh1M,EAAw1M,KAAx1M,EAAg2M,KAAh2M,EAAw2M,KAAx2M,EAAg3M,KAAh3M,EAAw3M,KAAx3M,EAAg4M,KAAh4M,EAAw4M,KAAx4M,EAAg5M,KAAh5M,EAAw5M,KAAx5M,EAAg6M,KAAh6M,EAAw6M,KAAx6M,EAAg7M,KAAh7M,EAAw7M,KAAx7M,EAAg8M,KAAh8M,EAAw8M,KAAx8M,EAAg9M,KAAh9M,EAAw9M,KAAx9M,EAAg+M,KAAh+M,EAAw+M,KAAx+M,EAAg/M,KAAh/M,EAAw/M,KAAx/M,EAAggN,KAAhgN,EAAwgN,KAAxgN,EAAghN,KAAhhN,EAAwhN,KAAxhN,EAAgiN,KAAhiN,EAAwiN,KAAxiN,EAAgjN,KAAhjN,EAAwjN,KAAxjN,EAAgkN,KAAhkN,EAAwkN,KAAxkN,EAAglN,KAAhlN,EAAwlN,KAAxlN,EAAgmN,KAAhmN,EAAwmN,KAAxmN,EAAgnN,KAAhnN,EAAwnN,KAAxnN,EAAgoN,KAAhoN,EAAwoN,KAAxoN,EAAgpN,KAAhpN,EAAwpN,KAAxpN,EAAgqN,KAAhqN,EAAwqN,KAAxqN,EAAgrN,KAAhrN,EAAwrN,KAAxrN,EAAgsN,KAAhsN,EAAwsN,KAAxsN,EAAgtN,KAAhtN,EAAwtN,KAAxtN,EAAguN,KAAhuN,EAAwuN,KAAxuN,EAAgvN,KAAhvN,EAAwvN,KAAxvN,EAAgwN,KAAhwN,EAAwwN,KAAxwN,EAAgxN,KAAhxN,EAAwxN,KAAxxN,EAAgyN,KAAhyN,EAAwyN,KAAxyN,EAAgzN,KAAhzN,EAAwzN,KAAxzN,EAAg0N,KAAh0N,EAAw0N,KAAx0N,EAAg1N,KAAh1N,EAAw1N,KAAx1N,EAAg2N,KAAh2N,EAAw2N,KAAx2N,EAAg3N,KAAh3N,EAAw3N,KAAx3N,EAAg4N,KAAh4N,EAAw4N,KAAx4N,EAAg5N,KAAh5N,EAAw5N,KAAx5N,EAAg6N,KAAh6N,EAAw6N,KAAx6N,EAAg7N,KAAh7N,EAAw7N,KAAx7N,EAAg8N,KAAh8N,EAAw8N,KAAx8N,EAAg9N,KAAh9N,EAAw9N,KAAx9N,EAAg+N,KAAh+N,EAAw+N,KAAx+N,EAAg/N,KAAh/N,EAAw/N,KAAx/N,EAAggO,KAAhgO,EAAwgO,KAAxgO,EAAghO,KAAhhO,EAAwhO,KAAxhO,EAAgiO,KAAhiO,EAAwiO,KAAxiO,EAAgjO,KAAhjO,EAAwjO,KAAxjO,EAAgkO,KAAhkO,EAAwkO,KAAxkO,EAAglO,KAAhlO,EAAwlO,KAAxlO,EAAgmO,KAAhmO,EAAwmO,KAAxmO,EAAgnO,KAAhnO,EAAwnO,KAAxnO,EAAgoO,KAAhoO,EAAwoO,KAAxoO,EAAgpO,KAAhpO,EAAwpO,KAAxpO,EAAgqO,KAAhqO,EAAwqO,KAAxqO,EAAgrO,KAAhrO,EAAwrO,KAAxrO,EAAgsO,KAAhsO,EAAwsO,KAAxsO,EAAgtO,KAAhtO,EAAwtO,KAAxtO,EAAguO,KAAhuO,EAAwuO,KAAxuO,EAAgvO,KAAhvO,EAAwvO,KAAxvO,EAAgwO,KAAhwO,EAAwwO,KAAxwO,EAAgxO,KAAhxO,EAAwxO,KAAxxO,EAAgyO,KAAhyO,EAAwyO,KAAxyO,EAAgzO,KAAhzO,EAAwzO,KAAxzO,EAAg0O,KAAh0O,EAAw0O,KAAx0O,EAAg1O,KAAh1O,EAAw1O,KAAx1O,EAAg2O,KAAh2O,EAAw2O,KAAx2O,EAAg3O,KAAh3O,EAAw3O,KAAx3O,EAAg4O,KAAh4O,EAAw4O,KAAx4O,EAAg5O,KAAh5O,EAAw5O,KAAx5O,EAAg6O,KAAh6O,EAAw6O,KAAx6O,EAAg7O,KAAh7O,EAAw7O,KAAx7O,EAAg8O,KAAh8O,EAAw8O,KAAx8O,EAAg9O,KAAh9O,EAAw9O,KAAx9O,EAAg+O,KAAh+O,EAAw+O,KAAx+O,EAAg/O,KAAh/O,EAAw/O,KAAx/O,EAAggP,KAAhgP,EAAwgP,KAAxgP,EAAghP,KAAhhP,EAAwhP,KAAxhP,EAAgiP,KAAhiP,EAAwiP,KAAxiP,EAAgjP,KAAhjP,EAAwjP,KAAxjP,EAAgkP,KAAhkP,EAAwkP,KAAxkP,EAAglP,KAAhlP,EAAwlP,KAAxlP,EAAgmP,KAAhmP,EAAwmP,KAAxmP,EAAgnP,KAAhnP,EAAwnP,KAAxnP,EAAgoP,KAAhoP,EAAwoP,KAAxoP,EAAgpP,KAAhpP,EAAwpP,KAAxpP,EAAgqP,KAAhqP,EAAwqP,KAAxqP,EAAgrP,KAAhrP,EAAwrP,KAAxrP,EAAgsP,KAAhsP,EAAwsP,KAAxsP,EAAgtP,KAAhtP,EAAwtP,KAAxtP,EAAguP,KAAhuP,EAAwuP,KAAxuP,EAAgvP,KAAhvP,EAAwvP,KAAxvP,EAAgwP,KAAhwP,EAAwwP,KAAxwP,EAAgxP,KAAhxP,EAAwxP,KAAxxP,EAAgyP,KAAhyP,EAAwyP,KAAxyP,EAAgzP,KAAhzP,EAAwzP,KAAxzP,EAAg0P,KAAh0P,EAAw0P,KAAx0P,EAAg1P,KAAh1P,EAAw1P,KAAx1P,EAAg2P,KAAh2P,EAAw2P,KAAx2P,EAAg3P,KAAh3P,EAAw3P,KAAx3P,EAAg4P,KAAh4P,EAAw4P,KAAx4P,EAAg5P,KAAh5P,EAAw5P,KAAx5P,EAAg6P,KAAh6P,EAAw6P,KAAx6P,EAAg7P,KAAh7P,EAAw7P,KAAx7P,EAAg8P,KAAh8P,EAAw8P,KAAx8P,EAAg9P,KAAh9P,EAAw9P,KAAx9P,EAAg+P,KAAh+P,EAAw+P,KAAx+P,EAAg/P,KAAh/P,EAAw/P,KAAx/P,EAAggQ,KAAhgQ,EAAwgQ,KAAxgQ,EAAghQ,KAAhhQ,EAAwhQ,KAAxhQ,EAAgiQ,KAAhiQ,EAAwiQ,KAAxiQ,EAAgjQ,KAAhjQ,EAAwjQ,KAAxjQ,EAAgkQ,KAAhkQ,EAAwkQ,KAAxkQ,EAAglQ,KAAhlQ,EAAwlQ,KAAxlQ,EAAgmQ,KAAhmQ,EAAwmQ,KAAxmQ,EAAgnQ,KAAhnQ,EAAwnQ,KAAxnQ,EAAgoQ,KAAhoQ,EAAwoQ,KAAxoQ,EAAgpQ,KAAhpQ,EAAwpQ,KAAxpQ,EAAgqQ,KAAhqQ,EAAwqQ,KAAxqQ,EAAgrQ,KAAhrQ,EAAwrQ,KAAxrQ,EAAgsQ,KAAhsQ,EAAwsQ,KAAxsQ,EAAgtQ,KAAhtQ,EAAwtQ,KAAxtQ,EAAguQ,KAAhuQ,EAAwuQ,KAAxuQ,EAAgvQ,KAAhvQ,EAAwvQ,KAAxvQ,EAAgwQ,KAAhwQ,EAAwwQ,KAAxwQ,EAAgxQ,KAAhxQ,EAAwxQ,KAAxxQ,EAAgyQ,KAAhyQ,EAAwyQ,KAAxyQ,EAAgzQ,KAAhzQ,EAAwzQ,KAAxzQ,EAAg0Q,KAAh0Q,EAAw0Q,KAAx0Q,EAAg1Q,KAAh1Q,EAAw1Q,KAAx1Q,EAAg2Q,KAAh2Q,EAAw2Q,KAAx2Q,EAAg3Q,KAAh3Q,EAAw3Q,KAAx3Q,EAAg4Q,KAAh4Q,EAAw4Q,KAAx4Q,EAAg5Q,KAAh5Q,EAAw5Q,KAAx5Q,EAAg6Q,KAAh6Q,EAAw6Q,KAAx6Q,EAAg7Q,KAAh7Q,EAAw7Q,KAAx7Q,EAAg8Q,KAAh8Q,EAAw8Q,KAAx8Q,EAAg9Q,KAAh9Q,EAAw9Q,KAAx9Q,EAAg+Q,KAAh+Q,EAAw+Q,KAAx+Q,EAAg/Q,KAAh/Q,EAAw/Q,KAAx/Q,EAAggR,KAAhgR,EAAwgR,KAAxgR,EAAghR,KAAhhR,EAAwhR,KAAxhR,EAAgiR,KAAhiR,EAAwiR,KAAxiR,EAAgjR,KAAhjR,EAAwjR,KAAxjR,EAAgkR,KAAhkR,EAAwkR,KAAxkR,EAAglR,KAAhlR,EAAwlR,KAAxlR,EAAgmR,KAAhmR,EAAwmR,KAAxmR,EAA

QACI,gB6EsC2B,yC;Q7ErC3B,MAAM,gCAAyB,SAAQ,WAAjC,C;;M6E0BV,Y;K;IAcA,gD;MAAA,iD;MAA2C,eAAK,eAAL,EAAAsB,GAAAtB,Q;MAA3C,Y;K;IAGA,yC;MAAA,iD;MAG8C,qB;MAC1C,KAAK,gBAAO,QAAP,C;MAJT,Y;K;IAqCJ,4B;MAK8E,gBAAnE,aAAmB,gEAAnB,C;MAA2E,wB;MAAlF,O1EvCO,S;K;;M2EjEP,uB;;kCAyCA,mB;MACI,UAAU,gBAAI,aAAI,OAAJ,EAAAa,IAAAb,C;MACd,OAAO,W;K;8BAGX,Y;MACI,gBAAI,Q;K;uCAOR,mB;MAA6D,OAAA,gBAAI,mBAAY,OAAZ,C;K;gCAEjE,Y;MAAyC,OAAA,gBAAI,U;K;iCAE7C,Y;MAAqD,OAAA,gBAAI,KAAK,W;K;qCAE9D,mB;MAAkD,OAAA,gBAAI,cAAO,OAAP,CAAJ,Q;K;+EAEpB,Y;MAAQ,OAAA,gBAAI,K;K;;IA5D1C,6B;MAAA,iD;MAGoB,8B;MAAZ,mB;MAAQ,oBAAM,gB;MAAd,yB;MAAA,oBAAM,Q;K;IAAN,O;MAAA,yC;MAAA,iD;MAG2C,8B;MAAnC,mB;MAAQ,oBAAM,oBAAN,gB;MAAd,yB;MAAA,oBAAM,Q;K;IAAN,O;MAAA,yC;MAAA,iD;MAGwC,8B;MAAhC,mB;MAAQ,oBAAM

OAAO,I;;QAGP,OAAO,Q;;K;+CAIf,e;MACI,IAAI,0BAAJ,C;QAAoB,OAAO,I;MAC3B,YAAY,kBAAW,GAAX,C;MACZ,IAAI,UAAU,SAAd,C;Q9BnDJ,O8BoDyB,iB9BpDnB,C8BoD+B,G9BpD/B,C;Q8BqDE,6B;QAEA,OAAO,K;;QAGP,OAAO,I;;K;wCAKf,Y;MACI,oBAAa,kB;MACb,YAAO,C;K;IAKA,0E;MAAA,oD;MACH,cAAkC,MAAAa,MAAK,mCAAL,C;MAC/C,kBAA4B,qBAAL,WAAK,C;MAC5C,kBAA+B,iBAA+B,I;K;iEAE/B,Y;MAAkC,OAAA,eAAS,U;K;8DAE3C,Y;MAIuB,gB;MAHnB,UAAU,eAAS,O;MACnB,iBAAU,G;MAES,+E;MAAnB,OAAO,iD;K;gEAGX,Y;MAEkC,UAA9B,M;MAAA,oC;MAA8B,YAAa,c;MjFchD,uB;MAeP,IAfoB,KAehB,QAAJ,C;QACI,cAhByB,0B;QAiBzB,MAAM,6BAAsB,OAAQ,WA9B,C;;QAEN,sBAAnB,KAAmB,C;K;;2CAjBnC,Y;MACI,yD;K;IAqBkD,0F;MAAA,8B;MAAA,oD;K;kHAAlD,Y;MAAkD,OAAA,uB;K;oHAAlD,Y;MAAkD,6CAAuB,gBAAvB,vB,C;K;2EAAlD,oB;MAAkD,wC,OAAA,2BAAuB,aAAvB,gBAAuB,EAAvB,QAAuB,C;K;qEAAlD,Y;MAAkD,OAAA,+B,OAAA,mCAAuB,gBAAvB,QAAuB,C;K;qEAAlD,Y;MAAkD,OAAA,mCAAuB,gBAAvB,QAAuB,C;K;mEAAlD,iB;MAAkD,gC

ACI,Y;MACI,mBAAM,IAAN,C;K;2CAGJ,mB;MACI,mBAAM,OAAN,C;MACA,c;K;iCAKJ,Y;K;;IAKuB,oC;MAA8B,qB;MAA7B,gC;K;2CACxB,mB;MAEI,oBA+DyC,OA/Dd,OA+Dc,C;MA9DzC,iBAAa,OAAM,aAAN,C;K;;IAIrB,8B;MAEoC,qB;K;iDAChC,mB;MACI,OAAQ,KAAI,OAAJ,C;K;mDAGZ,mB;MACI,OAAQ,KAAI,OAAJ,C;K;2CAGZ,Y;MACI,OAAQ,KAAI,EAAJ,C;K;;IAIhB,0B;MAEqC,qB;MACjC,cAAa,E;K;6CAEb,mB;MACI,eAoCyC,OApCxB,OAoCwB,C;K;qCAjC7C,Y;MACI,cAAS,E;K;;IAIjB,sC;MAE4C,yB;K;yDACxC,mB;MACI,QAwByC,OAxB1B,OAwB0B,C;MAvBzC,QAAQ,CxEqJoF,awErJhE,IxEqJgE,EwErJ1D,CxEqJ0D,C;MwEpJ5F,IAAI,KAAK,CAAT,C;QACI,4BAAU,CxE+J0E,WwE/J9D,CxE+J8D,EwE/J3D,CxE+J2D,C;QwE9JpF,Y;QACA,IAAI,CxE0JiE,WwE1JrD,IAAI,CAAJ,IxE0JqD,C;;MwExJzE,4BAAU,C;K;iDAGd,Y;MACI,OAAQ,KAAI,WAAJ,C;MACR,cAAS,E;K;;;IAIjB,yB;MACiD,cAAa,KAAb,C;K;IAEjD,mB;MAEI,MAAO,U;K;IAGX,4B;MAEI,MAAO,iBAAQ,OAAR,C;K;IAGX,wB;MAEI,MAAO,eAAM,OAAN,C;K;IAGX,kB;MACqC,MAAM,qCAA8B,sCAA9B,C;K;IAE3C,wB;MAC4C,MAAM,qCAA8B,4CAA9B,C;K;IAClGlD,mD;MACI,0B;MASA,gBAA2B,a;K;2FAEvB,Y;MAAQ,OAAA,eAAS,Q;K;oDAEjB,kB;MAAmB,UAAU,IAAK,S;MAEtC,YAAQ,2CAAR,C;MAAQ,OAAA,gBAAS,M;WAEjB,YAAQ,yBAAR,C;QAAQ,gBAAA,yCAAS,oBAAM,MAAX,C;;QAAW,OAAA,eAAS,6BAAT,iBAAS,C;;MAAT,OAAA,eAAS,M;WAEn

yB;MAAA,iB;MAAA,oB;QASmD,OAAA,MAAW,MAAK,CAAL,C;O;KAT9D,C;uEAWA,yB;MAAA,iB;MAAA,oB;QASmD,OAAA,MAAW,MAAK,CAAL,C;O;KAT9D,C;yEAWA,yB;MAAA,iB;MAAA,uB;QAkB+D,OAAA,MAAW,OAAM,CAAN,EAAS,CAAT,C;O;KAlB1E,C;uEAoBA,yB;MAAA,iB;MAAA,oB;QAUmD,OAAA,MAAW,MAAK,CAAL,C;O;KAV9D,C;uEAYA,yB;MAAA,iB;MAAA,oB;QASmD,OAAA,MAAW,MAAK,CAAL,C;O;KAT9D,C;uEAWA,yB;MAAA,iB;MAAA,oB;QAUmD,OAAA,MAAW,MAAK,CAAL,C;O;KAV9D,C;yEAYA,yB;MAAA,iB;MAAA,oB;QAYoD,OAAA,MAAW,OAAM,CAAN,C;O;KAZ/D,C;yEAcA,yB;MAAA,iB;MAAA,oB;QAYoD,OAAA,MAAW,OAAM,CAAN,C;O;KAZ/D,C;yEAcA,yB;MAAA,iB;MAAA,oB;QAaoD,OAAA,MAAW,OAAM,CAAN,C;O;KAb/D,C;yEAeA,yB;MAAA,iB;MAAA,uB;QAS+D,OAAA,MAAW,OAAM,CAAN,EAAS,CAAT,C;O;KAT1E,C;uEAWA,yB;MAAA,iB;MAAA,oB;QAQmD,OAAA,MAAW,MAAK,CAAL,C;O;KAR9D,C;qEAUA,yB;MAAA,iB;MAAA,oB;QAUkD,OAAA,MAAW,KAAI,CAAJ,C;O;KAV7D,C;yEAYA,yB;MAAA,iB;MAAA,oB;QAcoD,OAAA,MAAW,OAAM,CAAN,C;O;KAd/D,C;IAgBA,sB;MAcI,IAAI,QAAQ,GAAR,IAAe,sB;MAcI,IAAI,QAAQ,GAAR,IAAe,C;K;mEAAG,1C,yB;MAAA,iB;MAAA,oB;QAWiD,OAAA,MAAW,KAAI,CAAJ,C;O;KAX5D,C;yEAaA,yB;MAAA,iB;MAAA,oB;QAOoD,OAAA,MAAW,OAAM,CAAN,C;O;KAP/D,C;uEASA,yB;MAAA,iB;MAAA,oB;QAOmD,OAAA,MAAW,MAAK,CAAL,C;O;KAP9D,C;uEASA,yB;MAAA,iB;MAAA,oB;QAgBmD,OAAA,MAAW,OAAM,CAAN,C;O;KAhB9D,C;yEAkBA,yB;MAAA,iB;MAAA,oB;QAUkD,OAAA,MAAW,OAAM,CAAN,C;O;KAV7D,C;yEAYA,yB;MAAA,iB;MAAA,oB;QAUoD,OAAA,MAAW,OAAM,CAAN,C;O;KAV/D,C;+EAYA,yB;MAAA,iB;MAAA,oB;QAauD,OAAA,MAAW,OAAM,CAAN,C;O;KAblE,C;IAe

ASiD,OAAA,MAA8B,MAAZ,CAAY,C;O;KAT/E,C;yEAWA,yB;MAAA,iB;MAAA,oB;QAUiD,OAAA,MAA8B,MAAZ,CAAY,C;O;KAV/E,C;2EAYA,yB;MAAA,iB;MAAA,oB;QAYkD,OAAA,MAA+B,OAAZ,CAAY,C;O;KAZjF,C;2EAcA,yB;MAAA,iB;MAAA,oB;QAYkD,OAAA,MAA+B,OAAZ,CAAY,C;O;KAZjF,C;2EAcA,yB;MAAA,iB;MAAA,oB;QAakD,OAAA,MAA+B,OAAZ,CAAY,C;O;KAbjF,C;2EAeA,yB;MAAA,iB;MAAA,uB;QAS4D,OAAA,MA6C,OAA1B,CAA0B,B,EAAZ,CAAY,C;O;KATzG,C;yEAWA,yB;MAAA,iB;MAAA,oB;QAQiD,OAAA,MA8B,OAAZ,CAAY,C;O;KAR/E,C;uEAUA,yB;MAAA,iB;MAAA,oB;QAUgD,OAAA,MA6B,OAAZ,CAAY,C;O;KAV7E,C;2EAYA,yB;MAAA,iB;MAAA,oB;QAckD,OAAA,MA+B,OAAZ,CAAY,C;O;KAdjF,C;uEAgBA,yB;MAAA,mC;MAAA,0B;QAc6D,OAAA,mC,IAA7B,4B,CAA6B,B,EAAZ,CAAY,C;O;KAhBzG,C;qEAkBA,yB;MAAA,iB;MAAA,oB;QAW+C,OAAA,MA6B,OAAZ,CAAY,C;O;KAX3E,C;2EAaA,yB;MAAA,iB;MAAA,oB;QAakD,OAAA,MA+B,OAAZ,CAAY,C;O;KAbjF,C;2EAeA,yB;MAAA,iB;MAAA,oB;QAOiD,OAAA,MA+B,OAAZ,CAAY,C;O;KAP/E,C;yEASA,yB;MAAA,iB;MAAA,oB;QAOiD,OAAA,MA8B,OAAZ,CAAY,C;O;KAP/E,C;yEASA,yB;MAAA,iB;MAAA,oB;QAgBiD,OAAA,MA+B,OAAZ,CAAY,C;O;KAhB/E,C;yEAkBA,yB;MAAA,iB;MAAA,oB;QAUiD,OAAA,MA8B,OAAZ,CAAY,C;O;KAV/E,C;2EAYA,yB;MAAA,iB;MAAA,oB;QAUkD,OAAA,MA+B,OAAZ,CAAY,C;O;KAVjF,C;2EAYA,yB;MAAA,iB;MAAA,oB;QAUkD,OAAA,MA+B,OAAZ,CAAY,C;O;KAVjF,C;2EAYA,yB;MAAA,iB;MAAA,oB;QAWkD,OAAA,MA+B,OAAZ,CAAY,C;O;KAXjF,C;2EAaA,yB;MAAA,iB;MAAA,oB;QAWkD,OAAA,MA+B,OAAZ,CAAY,C;O;KAXjF,C;2EAaA,yB;MAAA,iB;MAAA,uB;QAQ4D,OAAA,MA6C,OAA1B,CAA0B,B,EAAZ,CAAY,C;O;KARzG,C;2EAUA,yB;MAAA,iB;MAAA,uB;QAQ4D,OAAA,MA6C,OAA1B,CAA0B,B,EAAZ,CAAY,C;O;KARzG,C;2EAUA,yB;MAAA,iB;MAAA,+B;QAasD,OAAA,MA8C,OAA1B,CAA0B,B,EAAZ,CAAY,C;O;KAbzG,C;2EAeA,yB;MAAA,iB;MAAA,+B;QAaoD,OAAA,MA8C,OAA1B,CAA0B,B,EAAZ,CAAY,C;O;KAbvG,C;2EAeA,yB;MAAA,iB;MAAA,4B;QAAA,OAAA,MAAA,MAAA,OAAA,MAAA,OAAA,MAAA,O

GqC,sBkDiGrC,C;QADlB,OlDjG4C,MAAa,KAAK,UAAS,GAAT,EkDkGvB,ClDlGuB,C;;QkDmGlD,aAAa,kBA
AL,IAAK,C;QAFzB,OlDjG4C,MAAa,KAAK,UkDmG7C,ClDnG6C,EAAc,MAAd,C;;K;IkDsGlE,uC;MAOU,UA
AU,SlD5GuC,a;MkD6GnD,c;QAAK,WAAa,iBlD5GkC,uBkD4GlC,C;QADtB,OlD7G4C,MAAa,KAAK,UkD8Gh
D,ClD9GgD,EAAc,IAAd,C;;QkD+GlD,YAAS,iBAAJ,GAAI,C;QAFrB,OlD7G4C,MAAa,KAAK,UAAS,KAAT,E
kD+GrB,ClD/GqB,C;;K;IkDkHlE,2C;MAaI,IAAI,CAAC,WAAa,EAAd,MAAqB,CAAzB,C;QACI,UAAU,SlD/Hy
C,a;QkDgInD,WAAW,SlD/HyC,c;QkDgIpD,aAAa,GAAI,IAAI,QAAR,GAAqB,IAAK,MAAK,CAAC,QAAD,IAA
L,C;QACvC,cAAc,IAAK,IAAI,QAAT,GAAsB,GAAI,MAAK,CAAC,QAAD,IAAL,C;QACxC,OAAW,CAAC,W
AAa,EAAd,MAAqB,CAAhC,GlDpIwC,MAAa,KAAK,UkDoIlB,MlDpIkB,EkDoIV,OlDpIU,CkDoI1D,GlDpIwC,
MAAa,KAAK,UkDoIS,OlDpIT,EkDoIkB,MlDpIlB,C;;QkDsInD,Q;QAAA,IAAI,CAAC,WAAa,EAAd,MAAqB,C
AAzB,C;UAAA,OAA4B,S;;uBlDpIiB,uB;UkDoIP,alDrIM,sB;UkDqI5C,OlDtIiC,MAAa,KAAK,kBAAc,MAAd,C;
;QkDsI1D,W;;K;kFAKR,yB;MAAA,4C;MAAA,sC;QAaiE,6BAAW,CAAC,QAAD,IAAX,C;O;KAbjE,C;qECvKA
,kC;MAII,OAAO,SAA8B,MAAK,WAAL,C;K;uEAGzC,8C;MAII,OAAO,SAA8B,MAAK,WAAL,EAAkB,UAAl
B,C;K;ICtCzC,iC;MACI,gBAAH,IAAI,OAAO,EAAG,GAAE,IAAI,IAAI,CAAC,CAAD,EAAI,EAAJ,CAAd,GAA
yB,CAAhC,C;K;;;IAKJ,sC;MACI,cAAO,QAAP,GAAkB,QAAQ,Q;K;ICP9B,yC;K;;IAWA,+B;K;;4GAYA,yB;MA
AA,gC;MAAA,yD;MAAA,sC;QAQI,OAAK,qBAAL,EAAK,SAAK,iB;O;KART,C;ICPI,2B;MAAS,Q;MAAD,OAAwB,C
AAvB,iEAAuB,Q;K;IAMhC,+B;MAAQ,iBAAU,SAAV,C;K;;;;;;;ICtB+B,4B;MACvC,8B;K;gEAAA,Y;MAAA,4B
;K;2FAII,Y;MzGO4B,MAAM,yB;K;kCyGLtC,iB;MACI,OAAO,oCAA0B,oBAAU,KAAM,OAAhB,C;K;oCAGrC
,Y;MAC+B,gB;MAAA,8FAA0B,C;K;oCAEzD,Y;MAEI,OAAO,oBAAQ,eAAR,C;K;;IAIyB,kC;MAAuB,sBAAc,
MAAd,C;MACL,Q;MAAtD,4BAAmC,CAAmB,OAAZ,MAAY,WAAnB,kC;K;8FAAnC,Y;MAAA,gC;K;oDAEA,
iB;MACW,cAAgB,W;MAAvB,OjEoEuD,MAAa,QiEpEpD,KjEoEoD,EAAY,OAAZ,C;K;;IiEhEjC,0E;MAIvC,sB
AAc,MAAd,C;MAFA,wC;MACA,8C;K;2CAEA,iB;MACI,IAAI,0CAAJ,C;QAAsC,OAAO,K;MAC7C,OAAa,uC
AAO,KAAP,CAAN,IAAuB,+BAAmB,KAAM,kBAAzB,C;K;iGAGD,Y;MAAQ,6B;K;uDAEzC,iB;MACI,OAAO,
0BAAmB,KAAnB,C;K;;IAIf,6B;MAAA,iC;MAAoC,sBAAoB,MAApB,C;MAChC,4BAAkC,S;K;+FAAlC,Y;MA
AA,gC;K;qDAEA,iB;MAAgD,Y;K;2FAG5C,Y;MAAQ,MAAM,qCAA8B,6CAA9B,C;K;yCAElB,iB;MAA4C,iBA
AU,I;K;2CAEtD,Y;MAA+B,Q;K;;;IAVnC,yC;MAAA,wC;QAAA,uB;;MAAA,iC;K;IAaA,uB;K;yFAqCqC,Y;M5G
0EY,MAAM,6B4G1EJ,oC5G0EkC,WAA9B,C;K;4F4GzEf,Y;M5GyES,MAAM,6B4GzED,uC5GyE+B,WAA9B,C
;K;+C4GvEnD,iB;M5GuE6C,MAAM,6B4GvEG,uC5GuE2B,WAA9B,C;K;mC4GrEnD,iB;MAA4C,iBAAU,I;K;q
CAEtD,Y;MAA+B,Q;K;;oHCnE/B,qB;MAAQ,2B;K;;;;;;;

E,gCAApE,C;MAEf,mBACkB,wBAAoB,MAApB,EAAgE,QAAhE,EAA0E,mCAA1E,C;MAElB,oBACmB,+B;MAEnB,oBACmB,wBAAoB,OAApB,EAAkE,SAAlE,EAA6E,oCAA7E,C;MAEnB,iBACgB,wBAAoB,MAApB,EAA8D,MAA9D,EAAsE,iCAAtE,C;MAEhB,kBACiB,wBAAoB,MAApB,EAA+D,OAA/D,EAAwE,kCAAxE,C;MAEjB,gBACe,wBAAoB,MAApB,EAA6D,KAA7D,EAAoE,gCAApE,C;MAEf,kBACiB,wBAAoB,MAApB,EAA+D,OAA/D,EAAwE,kCAAxE,C;MAEjB,mBACkB,wBAAoB,MAApB,EAAgE,QAAhE,EAA0E,mCAA1E,C;MAElB,kBACiB,wBAAoB,KAApB,EAAiE,OAAjE,EAA0E,kCAA1E,C;MAEjB,mBACkB,wBAAoB,MAApB,EAAgE,QAAhE,EAA0E,mCAA1E,C;MAElB,sBACqB,wBAAoB,KAApB,EAAkE,WAAlE,EAA+E,sCAA/E,C;MAErB,yBACwB,wBAAoB,KAApB,EAAkE,cAArE,EAAqF,yCAArF,C;MAExB,sBACqB,wBAAoB,WAApB,EAAsE,WAAtE,EAAmF,sCAAnF,C;MAErB,uBACsB,wBAAoB,UAApB,EAAwE,YAAxE,EAAsF,uCAAtF,C;MAEtB,qBACoB,wBAAoB,UAApB,EAAwE,UAAxE,EAAkF,qCAAlF,C;MAEpB,sBACqB,wBAAoB,KAApB,EAAkE,WAAlE,EAA+E,sCAA/E,C;MAErB,uBACsB,wBAAoB,YAApB,EAA0E,YAA1E,EAAwF,uCAAxF,C;MAEtB,wBACuB,wBAAoB,YAApB,EAA0E,aAA1E,EAAwF,wCAAxF,C;K;IAMkB,qE;MAAA,qB;QAAE,OvE/DD,OuE+DU,EAAT,KAAiB,UAAjB,IAAkC,EAAY,OAAf,KAA0B,a;O;K;+CAJpG,iB;MAAE,Q;MAAhB,U;MAAA,KAAgB,OAAhB,eAAgB,KAAhB,EAAgB,CAAhB,I,KAAgB,U;QAAA,a;;QAAhB,YAAgB,wBAAoB,QAApB,EAA8B,EAA9B,+DAAgB,kBAAhB,EAAgB,EAAhB,oBAAgB,mDAAhB,EAAgB,C;QAAhB,EAAgB,eAAhB,uBAAgB,qC

,EAG4B,IAH5B,C;MAUA,oC;K;;IA3JA,+C;MAAA,yB;MAAA,uC;K;;IAKA,qD;MAAA,yB;MAAA,6C;K;;IAKA
,qD;MAAA,yB;MAAA,6C;K;;IAKA,qD;MAAA,yB;MAAA,6C;K;;IAKA,oD;MAAA,yB;MAAA,4C;K;;IAKA,iD;
MAAA,yB;MAAA,yC;K;;IAKA,qD;MAAA,yB;MAAA,6C;K;;IAKA,mD;MAAA,yB;MAAA,2C;K;;IAKA,2D;M
AAA,yB;MAAA,mD;K;;IAKA,yD;MAAA,yB;MAAA,iD;K;;IAKA,kD;MAAA,yB;MAAA,0C;K;;IAKA,iD;MAA
A,yB;MAAA,yC;K;;IAKA,oD;MAAA,yB;MAAA,4C;K;;IAKA,mD;MAAA,yB;MAAA,2C;K;;IAKA,wD;MAAA,
yB;MAAA,gD;K;;IAKA,4C;MAAA,yB;MAAA,oC;K;;IAKA,2C;MAAA,yB;MAAA,mC;K;;IAKA,gD;MAAA,yB;
MAAA,wC;K;;IAKA,8C;MAAA,yB;MAAA,sC;K;;IAKA,qD;MAAA,yB;MAAA,6C;K;;IAKA,sD;MAAA,yB;MA
AA,8C;K;;IAKA,oD;MAAA,yB;MAAA,4C;K;;IAKA,0D;MAAA,yB;MAAA,kD;K;;IAKA,sD;MAAA,yB;MAAA,
8C;K;;IAKA,gD;MAAA,yB;MAAA,wC;K;;IAKA,oD;MAAA,yB;MAAA,4C;K;;IAKA,oD;MAAA,yB;MAAA,4C;
K;;IAKA,iD;MAAA,yB;MAAA,yC;K;;IAKA,8D;MAAA,yB;MAAA,sD;K;;IAKA,4D;MAAA,yB;MAAA,oD;K;8
CAKA,gB;MAG2D,OAAK,iBAAL,IAAK,CAAL,KAA2B,IAAK,c;K;IAE3F,kC;MAAA,sC;K;uDACI,oB;MAEQ,I
ADE,QACF,IAAG,CAAH,IADE,QACF,IAAM,EAAN,C;QADJ,OACgB,sBAAS,QAAT,C;WACZ,IAFE,QAEF,IA
AG,EAAH,IAFE,QAEF,IAAO,EAAP,C;QAFJ,OAEiB,sBAAS,WAAW,CAAX,IAAT,C;;QACL,MAAM,gCAAyB
,eAAY,QAAZ,qBAAzB,C;K;;;IAL1B,8C;MAAA,yB;MAAA,6C;QAAA,4B;;MAAA,sC;K;;IA7JJ,+B;MAAA,+yC
;K;;IAAA,oC;MAAA,a;aAAA,Y;UAAA,4C;aAAA,kB;UAAA,kD;aAAA,kB;UAAA,kD;aAAA,kB;UAAA,kD;aA
AA,iB;UAAA,iD;aAAA,c;UAAA,8C;aAAA,kB;UAAA,kD;aAAA,gB;UAAA,gD;aAAA,wB;UAAA,wD;aAAA,sB
;UAAA,sD;aAAA,e;UAAA,+C;aAAA,c;UAAA,8C;aAAA,iB;UAAA,iD;aAAA,gB;UAAA,gD;aAAA,qB;UAAA,q
D;aAAA,S;UAAA,yC;aAAA,Q;UAAA,wC;aAAA,a;UAAA,6C;aAAA,W;UAAA,2C;aAAA,kB;UAAA,kD;aAAA,
mB;UAAA,mD;aAAA,iB;UAAA,iD;aAAA,uB;UAAA,uD;aAAA,mB;UAAA,mD;aAAA,a;UAAA,6C;aAAA,iB;U
AAA,iD;aAAA,iB;UAAA,iD;aAAA,c;UAAA,8C;aAAA,2B;UAAA,2D;aAAA,yB;UAAA,yD;;UAAA,6D;;K;;ICKi
D,2C;uBAA+B,O;;K;;IAC5E,8C;MAAA,kE;MAAuB,qCAAK,IAAL,C;MAAvB,Y;K;ICD8B,gC;MAe9B,gBAAiC
,YAAY,SAAhB,GAA2B,OAA3B,GAAwC,E;K;uFAGjE,Y;MAAQ,OAAO,aAAY,O;K;yCAE/B,iB;MACW,gBAA
P,a;MrGoGG,Q;MAAA,IqGpGc,KrGoGV,IAAS,CAAT,IqGpGU,KrGoGI,IAAS,2BAA3B,C;QAAA,OAAsC,qBq
GpGxB,KrGoGwB,C;;QqGpGf,MAAM,8BAA0B,mCAAyB,WAAzB,MAA1B,C;;MAAhC,W;K;kDAEJ,gC;MAA
gF,OAAA,a1GiMY,W0GjMK,U1GiML,E0GjMiB,Q1GiMjB,C;K;6C0G/L5F,iB;MACI,qCAAU,KAAV,C;MACA,
OAAO,I;K;6CAGX,iB;MACI,iBAAgB,SAAN,KAAM,C;MAChB,OAAO,I;K;6CAGX,uC;MACI,OAAA,IAAK,q
BAAY,wBAAS,MAArB,EAA6B,UAA7B,EAAyC,QAAzC,C,C;K;sCAET,Y;MAayB,UAEK,M;MAL1B,eAAe,E;M
ACf,YAAY,aAAO,OAAP,GAAgB,CAAhB,I;MACZ,OAAO,SAAS,CAAhB,C;QACI,UAAU,0BAAO,YAAP,EAA
O,oBAAP,Q;QACV,IAAQ,eAAJ,GAAI,CAAJ,IAAwB,SAAS,CAArC,C;UACI,WAAW,0BAAO,cAAP,EAAO,sB
AAP,U;UACX,IAAS,gBAAL,IAAK,CAAT,C;YACI,WAAW,+BAAW,iBAAX,wBAAkB,gBAAlB,C;;YAEX,WA
AW,+BAAW,gBAAX,wBAAiB,iBAAjB,C;;;UAGf,gCAAY,GAAZ,C;;;MAGR,gBAAS,Q;MACT,OAAO,I;K;6CA
GX,iB;MAOI,iBAAgB,SAAN,KAAM,C;MAChB,OAAO,I;K;6CAGX,iB;MAQI,iBAAU,K;MACV,OAAO,I;K;6C
AGX,iB;MAQI,iBAAgB,eAAN,KAAM,C;MAChB,OAAO,I;K;6CAGX,iB;MAC2C,2BAAO,KAAP,C;K;6CAE3C
,iB;MAOI,gBAAA,IAAK,SAAL,IAAe,wBAAS,MAAxB,C;MACA,OAAO,I;K;uCAGX,Y;MAU6B,kB;K;qDAE7
B,2B;K;8CAcA,kB;MAO0C,OAAA,IAAY,SAAY,SAAQ,MAAR,C;K;8CAElE,8B;MAQ2D,OAAA,IAAY,SAAY,
SAAQ,MAAR,EAAAgB,UAAhB,C;K;kDAEnF,kB;MAQ8C,OAAA,IAAY,SAAY,aAAY,MAAZ,C;K;kDAEtE,8B;
MASI,IAAI,MpGuGwC,YAAU,CoGvGlD,IAAoB,aAAa,CAArC,C;QAAwC,OAAO,E;MAC/C,OAAO,IAAY,SA
AY,aAAY,MAAZ,EAAoB,UAApB,C;K;4CAGnC,wB;MAWI,oCAAa,4BAAmB,KAAnB,EAA0B,WAA1B,C;MA
Eb,gBAAS,a1GmB+E,W0GnB9D,C1GmB8D,E0GnB3D,K1GmB2D,C0GnB/E,YAA6B,KAA7B,IAAqC,a1GgB2
B,W0GhBV,K1GgBU,C;M0GfzE,OAAO,I;K;6CAGX,wB;MAQI,oCAAa,4BAAmB,KAAnB,EAA0B,WAA1B,C;
MAEb,gBAAS,a1GK+E,W0GL9D,C1GK8D,E0GL3D,K1GK2D,C0GL/E,uBAA6B,kBAA7B,IAAqC,a1GE2B,W0
GFV,K1GEU,C;M0GDzE,OAAO,I;K;6CAGX,wB;MAUI,oCAAa,4BAAmB,KAAnB,EAA0B,WAA1B,C;MAEb,g
BAAS,a1GX+E,W0GW9D,C1GX8D,E0GW3D,K1GX2D,C0GW/E,GAAmC,eAAN,KAAM,CAAnC,GAAsD,a1G
dU,W0GcO,K1GdP,C;M0GezE,OAAO,I;K;6CAGX,wB;MAaI,oCAAa,4BAAmB,KAAnB,EAA0B,WAA1B,C;MA
Eb,gBAAS,a1G9B+E,W0G8B9D,C1G9B8D,E0G8B3D,K1G9B2D,C0G8B/E,GAAmC,SAAN,KAAM,CAAnC,GA
AgD,a1GjCgB,W0GiCC,K1GjCD,C;M0GkCzE,OAAO,I;K;6CAGX,wB;MAWI,oCAAa,4BAAmB,KAAnB,EAA0
B,WAA1B,C;MAEb,gBAAS,a1G/C+E,W0G+C9D,C1G/C8D,E0G+C3D,K1G/C2D,C0G+C/E,GAAmC,SAAN,KA
AM,CAAnC,GAAgD,a1GlDgB,W0GkDC,K1GlDD,C;M0GmDzE,OAAO,I;K;6CAGX,wB;MACuD,2BAAO,KAA

P,EAAc,KAAd,C;K;6CAEvD,wB;MAUI,oCAAa,4BAAmB,KAAnB,EAA0B,WAA1B,C;MAEb,eAAe,wBAAS,M;MACxB,gBAAc,IAAK,S1GnEqE,W0GmEpD,C1GnEoD,E0GmEjD,K1GnEiD,C0GmE1E,GAAkC,QAAlC,GAA6C,IAAK,S1GtES,W0GsEQ,K1GtER,C;M0GuEzE,OAAO,I;K;gDAGX,qB;MAcI,IAAI,YAAY,CAAhB,B,C;QACI,MAAM,gCAAY,B,0BAAA,uBAAS,SAAvB,I,C,wBAAA,C;;MAGV,IAAI,aAAa,WAAjB,B,C;QACI,gBAAS,a1G1F2E,W0G0F1D,C1G1F0D,E0G0FvD,S1G1FuD,C;;Q0G4FpF,aAAU,WAAV,MAAuB,SAAvB,M;UACI,qCAAU,CAAV,C;;;K;gDAGX,sB;MAQI,oCAAa,4BAAmB,UAAnB,EAA+B,WAA/B,C;MAEb,OAAO,a1G/GkE,W0G+GjD,U1G/GiD,C;K;gD0GkH7E,gC;MAQI,oCAAa,4BAAmB,UAAnB,EAA+B,QAA/B,EAAyC,WAAzC,C;MAEb,OAAO,a1GzHiF,W0GyHhE,U1GzHgE,E0GyHpD,Q1GzHoD,C;K;yC0G4H5F,Y;K;uCAcA,Y;MAAkC,oB;K;oCAAlC,Y;MAOI,gBAAS,E;MACT,OAAO,I;K;0CAGX,wB;MAQI,oCAAa,2BAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,gBAAS,a1GjK+E,W0GiK9D,C1GjK8D,E0GiK3D,K1GjK2D,C0GiK/E,uBAA6B,kBAA7B,IAAqC,a1GpK2B,W0GoKV,QAAlC,EAA4C,QAA5C,EAAsD,CAAtD,EAAyD,I1GpKuB,C0GoKpB,C;K;+C0GuKpE,uC;MAQI,oCAAa,4BAAmB,UAAnB,EAA+B,QAA/B,EAAyC,WAAzC,C;MAEb,OAAO,a1GjLkE,W0GiLjD,U1GjLiD,C0GiL7D,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GrLuB,C0GqLpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GhNkE,W0GgNjD,U1GhNiD,C0GgN7D,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GpNuB,C0GoNpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G/OkE,W0G+OjD,U1G/OiD,C0G+O7D,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GnPuB,C0GmPpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G9QkE,W0G8QjD,U1G9QiD,C0G8Q7D,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GlRuB,C0GkRpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G7SkE,W0G6SjD,U1G7SiD,C0G6S7D,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GjTuB,C0GiTpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G5UkE,W0G4UjD,U1G5UiD,C0G4U7D,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GhVuB,C0GgVpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G3WkE,W0G2WjD,U1G3WiD,C0G2W7D,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G/WuB,C0G+WpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G1YkE,W0G0YjD,U1G1YiD,C0G0Y7D,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G9YuB,C0G8YpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GzakE,W0GyajD,U1GzaiD,C0GyaxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G7auB,C0G6apB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GxbkE,W0GwbjD,U1GxbiD,C0GwbxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G5buB,C0G4bpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GvckE,W0GucjD,U1GvciD,C0GucxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G3cuB,C0G2cpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GtdkE,W0GsdjD,U1GtdiD,C0GsdxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G1duB,C0G0dpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GreiE,W0GqehD,U1GreiD,C0GqexD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GzeuB,C0GyepB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GpfiE,W0GofhD,U1GpfiD,C0GofxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GxfuB,C0GwfpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GngBiE,W0GmgBhD,U1GngBiD,C0GmgBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GvguB,C0GugBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GlhBiE,W0GkhBhD,U1GlhBiD,C0GkhBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GthBuB,C0GshBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GjiBiE,W0GiiBhD,U1GjiBiD,C0GiiBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GriBuB,C0GqiBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GpjBiE,W0GojBhD,U1GpjBiD,C0GojBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GxjBuB,C0GwjBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GvkBiE,W0GukBhD,U1GvkBiD,C0GukBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G3kBuB,C0G2kBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G1lBiE,W0G0lBhD,U1G1lBiD,C0G0lBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G9luB,C0G8lBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G7mBiE,W0G6mBhD,U1G7mBiD,C0G6mBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GjnBuB,C0GinBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GhoBiE,W0GgoBhD,U1GhoBiD,C0GgoBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GpouB,C0GooBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GnpBiE,W0GmpBhD,U1GnpBiD,C0GmpBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GvpBuB,C0GupBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GtqBiE,W0GsqBhD,U1GtqBiD,C0GsqBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G1quB,C0G0qBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GzrBiE,W0GyrBhD,U1GzrBiD,C0GyrBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G7ruB,C0G6rBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G5sBiE,W0G4sBhD,U1G5sBiD,C0G4sBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GhtBuB,C0GgtBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G3tBiE,W0G2tBhD,U1G3tBiD,C0G2tBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G/tBuB,C0G+tBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G1uBiE,W0G0uBhD,U1G1uBiD,C0G0uBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G9uBuB,C0G8uBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GzvBiE,W0GyvBhD,U1GzvBiD,C0GyvBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G7vBuB,C0G6vBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GxwBiE,W0GwwBhD,U1GxwBiD,C0GwwBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G5wBuB,C0G4wBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GvxBiE,W0GuxBhD,U1GvxBiD,C0GuxBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G3xBuB,C0G2xBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GtyBiE,W0GsyBhD,U1GtyBiD,C0GsyBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G1yBuB,C0G0yBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GrzBiE,W0GqzBhD,U1GrzBiD,C0GqzBxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GzzBuB,C0GyzBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G10BiE,W0G00BhD,U1G10BiD,C0G00BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G90BuB,C0G80BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G31BiE,W0G21BhD,U1G31BiD,C0G21BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G/1BuB,C0G+1BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G52BiE,W0G42BhD,U1G52BiD,C0G42BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1Gh3BuB,C0Gg3BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G73BiE,W0G63BhD,U1G73BiD,C0G63BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1Gj4BuB,C0Gi4BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G94BiE,W0G84BhD,U1G94BiD,C0G84BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1Gl5BuB,C0Gk5BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G/5BiE,W0G+5BhD,U1G/5BiD,C0G+5BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1Gn6BuB,C0Gm6BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1Gh7BiE,W0Gg7BhD,U1Gh7BiD,C0Gg7BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1Gp7BuB,C0Go7BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1Gj8BiE,W0Gi8BhD,U1Gj8BiD,C0Gi8BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1Gr8BuB,C0Gq8BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1Gl9BiE,W0Gk9BhD,U1Gl9BiD,C0Gk9BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1Gt9BuB,C0Gs9BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1Gn+BiE,W0Gm+BhD,U1Gn+BiD,C0Gm+BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1Gv+BuB,C0Gu+BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1Gp/BiE,W0Go/BhD,U1Gp/BiD,C0Go/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1Gx/BuB,C0Gw/BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1Gr/BiE,W0Gq/BhD,U1Gr/BiD,C0Gq/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1Gz/BuB,C0Gy/BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1Gt/BiE,W0Gs/BhD,U1Gt/BiD,C0Gs/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G1/BuB,C0G0/BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1Gv/BiE,W0Gu/BhD,U1Gv/BiD,C0Gu/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G3/BuB,C0G2/BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1Gx/BiE,W0Gw/BhD,U1Gx/BiD,C0Gw/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G5/BuB,C0G4/BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1Gz/BiE,W0Gy/BhD,U1Gz/BiD,C0Gy/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G7/BuB,C0G6/BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G1/BiE,W0G0/BhD,U1G1/BiD,C0G0/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G9/BuB,C0G8/BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G3/BiE,W0G2/BhD,U1G3/BiD,C0G2/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G//BuB,C0G+/BpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G5/BiE,W0G4/BhD,U1G5/BiD,C0G4/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GhguB,C0GggBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G7/BiE,W0G6/BhD,U1G7/BiD,C0G6/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GjguB,C0GigBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G9/BiE,W0G8/BhD,U1G9/BiD,C0G8/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GlguB,C0GkgBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G//BiE,W0G+/BhD,U1G//BiD,C0G+/BxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GnguB,C0GmgBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GhgCiE,W0GggChD,U1GhgCiD,C0GggCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GpguB,C0GogBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GjgCiE,W0GigChD,U1GjgCiD,C0GigCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GrguB,C0GqgBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GlgCiE,W0GkgChD,U1GlgCiD,C0GkgCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GtguB,C0GsgBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GngCiE,W0GmgChD,U1GngCiD,C0GmgCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GvguB,C0GugBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GpgCiE,W0GogChD,U1GpgCiD,C0GogCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GxguB,C0GwgBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GrgCiE,W0GqgChD,U1GrgCiD,C0GqgCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GzguB,C0GygBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GtgCiE,W0GsgChD,U1GtgCiD,C0GsgCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G1guB,C0G0gBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GvgCiE,W0GugChD,U1GvgCiD,C0GugCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G3guB,C0G2gBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GxgCiE,W0GwgChD,U1GxgCiD,C0GwgCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G5guB,C0G4gBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1GzgCiE,W0GygChD,U1GzgCiD,C0GygCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G7guB,C0G6gBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G1gCiE,W0G0gChD,U1G1gCiD,C0G0gCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G9guB,C0G8gBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G3gCiE,W0G2gChD,U1G3gCiD,C0G2gCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1G/guB,C0G+gBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kDAGX,gC;MAQI,oCAAa,2BAAkB,UAAlB,EAA8B,WAA9B,C;MAEb,OAAO,a1G5gCiE,W0G4gChD,U1G5gCiD,C0G4gCxD,GAAuC,KAAvC,EAAiD,GAAjD,C,IAAsD,S1GhhuB,C0GghBpB,O,W0GqGrE,qB,Q1GrGqE,Q0GqGrE,C;M0GsFI,yBAAkB,KAAlB,EAAyB,WAAzB,C;MAEb,OAAO,I;K;kD

AR,iCAAoB,CAAQ,kBAAK,EAAL,CAAR,6BAAxB,C;QACI,OAAO,I;;MAEX,IAAI,YAAO,GAAX,C;QACI,OAAO,K;;MAEX,OAAO,uB;K;IAGX,oC;MAUI,IAAI,CAAQ,kBAAK,GAAL,CAAR,iCAAoB,CAAQ,kBAAK,EAAL,CAAR,6BAApB,IAAAwC,CAAQ,kBAAK,EAAL,CAAR,6BAA5C,C;QACI,OAAO,I;;MAEX,IAAI,YAAO,GAAX,C;QACI,OAAO,K;;MAGX,OAAO,0BAAiB,uB;K;IAG5B,4B;MASI,IAAI,CAAQ,kBAAK,EAAL,CAAR,6BAAJ,C;QACI,OAAO,I;;MAEX,IAAI,YAAO,GAAX,C;QACI,OAAO,K;;MAEX,OAAO,sB;K;IAGX,gC;MAUI,IAAI,CAAQ,kBAAK,GAAL,CAAR,6BAAJ,C;QACI,OAAO,I;;MAEX,IAAI,YAAO,GAAX,C;QACI,OAAO,K;;MAEX,OAAO,0B;K;IAGX,gC;MASI,IAAI,YAAO,GAAX,C;QACI,OAAO,K;;MAEX,OAAO,gCAAoD,yD;K;IAG/D,iC;MAUI,OAAO,aAAQ,EAAR,IAAAoB,CAAQ,mBAAU,GAAV,CAAR,6B;K;IAGX,iC;MAUI,OAAO,wDAAiB,kBAAK,SAAL,CAAjB,6B;K;IAGX,iE

AkB,C;MAC1D,IAAI,aAAa,CAAb,IAAkB,aAAa,KAAM,OAAzC,C;QACI,MAAM,8BAA0B,gCAA6B,UAA7B,wBAAyD,KAAM,OAAzF,C;;MAEV,OAAqB,SAAd,oBAAc,EAAS,KAAM,WAAf,EAA2B,UA3B,EAAuC,oBAAvC,C;K;IAeG,6E;MAAA,mB;QAAE,+BAAK,aAAL,EAAY,kBAAZ,C;O;K;IA2B,uC;MAAW,OAAA,KAAM,O;K;sCAZ1E,6B;MAQ+C,0B;QAAA,aAAkB,C;MAC7D,IAAI,aAAa,CAAb,IAAkB,aAAa,KAAM,OAAzC,C;QACI,MAAM,8BAA0B,gCAA6B,UAA7B,wBAAyD,KAAM,OAAzF,C;;MAEV,OAAO,mBAAiB,6CAAjB,EAAA8C,sBAAA,OAAA,mBAAA,EAAA,OAAA,KAAA,OAAA,KAAA,OAAA,K

AI,2BAAJ,C;QACI,yH;;MAKJ,OAAO,kC;K;4CAGf,Y;MACI,OAAY,SAAZ,wBAAY,EAAS,kBAAT,EAAoB,kBAAM,UAAV,GAAAqB,8BAAAuB,kBAAM,MAA7B,CAArB,GAA8D,kBAAM,aAAN,GAAAqB,CAArB,IAA9E,EAAAsG,wBAAAtG,C;K;gEAAhB,iB;MACI,IAAI,QAAc,iBAAN,kBAAM,CAAlB,C;QACI,YAAkB,kBAAY,YAAW,KAAAX,C;QAC9B,IAAa,KAAT,sBAAAiB,KAAAR,rB,C;UACI,YAAkB,kBAAY,YAAW,QAAQ,CAAR,IAAX,IAAa,KAAT,sBAAAiB,KAAAR,rB,C;YACI,OAAO,QAAQ,CAAR,IAAP,I;;;;MAAnB,OAAO,QAAQ,CAAR,IAAP,I;K;IAAjCiC,oE;MAAAA,kC;MAAA+B,6B;K;mHAAAhD,Y;MAAQ,OAAR,QAAQ,kBAAM,O;K;IACqC,4E;MAAA,qB;QAAA,yBAAAkC,C;O;K;qEAAA5E,Y;MAAA,iDAAA,qBAAA,oBAAA,aAAA,oBAAA,eAAA,iBAAA,eAAA,iBAAA,W;K;wEAAA,iB;MAAA,4C,Q;MAAA,eAAA,kBAAA,sCAAA,sCAAA,sCAAA,sCAAA,oBAAA,oB;QAAA,yB,O;K;;;IAAA,iDAAA,uD;MAAA,sBAAA,iB,I;MAAA,iB,B,Y,AAAA,e,kB,K,C;MAAA,2B,C,Q;MAAA,eAAA,kBAAA,kBAAA,kB

MAAO,KSHM,ETGN,ESHU,ETGV,C;QSFV,IAAI,QAAO,CAAX,C;UAAc,OAAO,KAAK,EAAL,I;QACrB,iBA
Ac,CAAd,UAAsB,GAAtB,U;UACI,eAAe,qBAAK,KAAL,C;UACf,gBAAgB,iBAAM,KAAN,C;UAEhB,IAAI,aA
AY,SAAhB,C;YACI,WAAoB,cAAT,QAAS,C;YACpB,YAAsB,cAAV,SAAU,C;YAEtB,IAAI,aAAY,SAAhB,C;c
ACwB,kBAAT,Q;cAAX,WDlO2C,gCAAY,cAfrB,YAAY,CAAZ,C;cCkPZ,kBAAV,S;cAAZ,YDnO2C,gCAAY,c
AfrB,YAAY,CAAZ,C;cCoPlC,IAAI,aAAY,SAAhB,C;gBACI,OAAgB,iBAAT,QAAS,EAAU,SAAV,C;;;;;QAKhC
,OAAO,KAAK,EAAL,I;;QAEP,OAAO,4BAAU,KAAV,C;;K;IAIf,4C;MAOqF,oCAAkB,KAAlB,C;K;IAErF,wD;
MASI,OAAW,UAAJ,GACE,4BAAL,SAAK,EAA4B,KAA5B,CADF,GAGE,kBAAL,SAAK,EAAkB,KAAlB,C;K;
IAIkD,oD;MAAU,OAAE,UAAF,CAAE,EAAU,CAAV,EAA0B,IAA1B,C;K;;IAIvE,+C;MAAQ,oC;K;2F8G/SZ,oC
;MACiF,O9G2Me,kB8G3ME,oBAAH,EAAG,C9G2MF,E8G3Mc,S9G2Md,C;K;mG8GzMhG,oC;MACqF,O9G2M
e,sB8G3MM,oBAAH,EAAG,C9G2MN,E8G3MkB,S9G2MlB,C;K;I8GzMpG,mD;MAIoD,0B;QAAA,aAAsB,K;M
ACtE,IAAI,CAAC,UAAL,C;QACI,O9GsMqF,qB8GtM7D,M9GsM6D,E8GtMrD,C9GsMqD,C;;Q8GpMrF,OAAO,
yBAAc,CAAd,EAAiB,MAAjB,EAAyB,CAAzB,EAA4B,MAAO,OAAnC,EAA2C,EAA2C,UAA3C,C;K;IAGf,iE;MAIqE,0
B;QAAA,aAAsB,K;MACvF,IAAI,CAAC,UAAL,C;QACI,O9G2LqF,qB8G3L7D,M9G2L6D,E8G3LrD,U9G2LqD,
C;;Q8GzLrF,OAAO,yBAAc,UAAd,EAA0B,MAA1B,EAAkC,CAAlC,EAAqC,MAAO,OAA5C,EAAoD,UAApD,
C;K;IAGf,iD;MAIkD,0B;QAAA,aAAsB,K;MACpE,IAAI,CAAC,UAAL,C;QACI,O9GmLoE,mB8GnL9C,M9GmL8C,C;;Q8GjLpE,OAAO,yBAAc,mBAAS,MAAO,OAAhB,IAAgd,EAAsC,MAAtC,EAA8C,CAA9C,EAAiD,MAAO
O,OAAxD,EAAgE,UAAhE,C;K;IAGf,mC;MAGI,aACa,S9G0L2D,O8G1LhD,K9G0LgD,C;M8GzLxE,OAAO,kB
AAkB,MAAO,OAAP,KAAe,C;K;IAG5C,4B;MAKoD,gCAAU,C;MAAV,U;QAAuB,kBAAR,yB;QAAQ,c;;UrH2n
DvD,U;UADhB,IAAI,0CAAsB,qBAA1B,C;YAAqC,aAAO,I;YAAP,e;;UACrB,+B;UAAhB,OAAgB,gBAAhB,C;Y
AAgB,2B;YAAM,IAAI,CqH3nD4D,aAAT,qBrH2nDxC,OqH3nDwC,CAAS,CrH2nDhE,C;cAAyB,aAAO,K;cAAP
,e;;;UAC/C,aAAO,I;;;QqH5nDgE,iB;;MAAvB,W;K;IAEpD,gD;MASiD,0B;QAAA,aAAsB,K;MAOxC,Q;MAN3B,
IAAI,iBAAJ,C;QAAkB,OAAO,a;MACzB,IAAI,aAAJ,C;QAAmB,OAAO,K;MAC1B,IAAI,CAAC,UAAL,C;QAAiB
B,OAAO,kBAAQ,KAAR,C;MAExB,IAAI,SAAK,OAAL,KAAe,C;QAAiC,OAAO,K;MAEb,OAAO,K;MAEb,OA
AL,SAAK,O;MAA3B,iBAAc,CAAd,wB;QACI,eAAe,qBAAK,KAAL,C;QACf,gBAAgB,iBAAM,KAAN,C;QAChB,IAAI,CAAU,SAAT,QAAS,EAAO,SAAP,C;UACI,OAAO,K;;;MAIf,OAAO,I;K;IAIX,sF
;MACkH,0B;QAAA,aAAsB,K;MACpI,oCAAkB,UAAlB,EAA8B,EAA9B,EAAiC,KAAjC,EAAwC,WAAxC,C;QAAqD,kD,MAAlD,EA
A0D,UAA1D,C;K;IAGJ,+B;MAYI,OxGmMmD,mBAAS,CwGnM5D,G9GwH4F,oB8GxHzD,C9GwHyD,E8GxHt
D,C9GwHsD,CAvC9B,c8GjFrC,G9GqHoD,oB8GrHZ,C9GqHY,C8GrH7E,GAAyE,S;K;IAG7E,iC;MASI,OxGuL
mD,mBAAS,CwGvL5D,G9G4G4F,oB8G5GzD,C9G4GyD,E8G5GtD,C9G4GsD,CAlB9B,c8G1FrC,G9GyGoD,oB
8GzGZ,C9GyGY,C8GzG7E,GAAyE,S;K;IAG7E,8B;MAOiB,IAAN,I;M3H/FP,IAAI,E2H8FI,KAAK,C3H9FT,CA
AJ,C;QACI,c2H6Fc,oD;Q3H5Fd,MAAM,gCAAyB,OAAQ,WAAjC,C;;M2H6FH,QAAM,CAAN,C;aACH,C;UAA
K,S;UAAL,K;aACA,C;UAAU,OAAL,SAAK,W;UAAV,K;;UAEI,aAAa,E;UACb,IAAI,ExGgKoC,qBAAU,CwGh
K9C,CAAJ,C;YACI,QAAQ,SAAK,W;YACb,YAAY,C;YACZ,OAAO,IAAP,C;cACI,IAAI,CAAC,QAAU,CAAV,EAAa,MAAb,C;MAIiB,OAAO,K;MAAP,IAAb,MAAa,C;QAAb,IAAI,CAAC,UAAL,C;UAAiB,OAAO,K;;MAIrB,OAAO,I;K;IAIX,sF
;MAAqE,0B;QAAA,aAAsB,K;MACvF,IAAI,CAAC,UAAL,C;QACI,O9GkKoE,mB8GlK9C,M9GkK8C,C;;Q8GhKpE,OAAO,yBAAc,mBAAS,MAAO,OAAhB,IAAgd,EAAsC,MAAtC,EAA8C,CAA9C,EAAiD,MAAO
O,OAAxD,EAAgE,UAAhE,C;K;IAGf,mC;MAGI,aACa,S9GyJ2D,O8GzJhD,K9GyJgD,C;M8GxJ

U,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAEX,YAAY,KAAa,CAAP,KAAO,C;MACzB,IAAI,SAAU,GAAV,
MAAkB,GAAtB,C;QACI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAEX,OAAQ,SAAU,CAAX
,GAAkB,KAAlB,GAA4B,I;K;IAGvC,yE;MASI,IAAI,SAAS,QAAb,C;QACI,OAAO,UAAU,CAAV,EAAa,KAAb,
EAAoB,gBAApB,C;;MAGX,YAAY,KAAa,CAAP,KAAO,C;MACzB,IAAI,SAAU,EAAV,MAAiB,CAArB,C;QA
CI,IAAI,SAAU,GAAV,MAAkB,GAAtB,C;UAEI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;aAER
,IAAI,SAAU,EAAV,MAAiB,EAArB,C;QACH,IAAI,SAAU,GAAV,MAAkB,GAAtB,C;UAEI,OAAO,UAAU,CA
AV,EAAa,KAAb,EAAoB,gBAApB,C;;aAER,IAAI,SAAU,GAAV,MAAkB,GAAtB,C;QACH,OAAO,UAAU,CAA
V,EAAa,KAAb,EAAoB,gBAApB,C;;MAGX,IAAI,SAAQ,CAAR,UAAa,QAAjB,C;QACI,OAAO,UAAU,CAAV,
EAAa,KAAb,EAAoB,gBAApB,C;;MAEX,YAAY,KAAiB,CAAX,KAAW,C;MAC7B,IAAI,SAAU,CAAV,MAAiB,CAAr
B,C;UAEI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;aAER,IAAI,SAAU,EAAV,MAAiB,CAArB,C;QA
CH,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAEX,IAAI,SAAQ,CAAR,UAAa,QAAjB,C;QACI,OAAO,UAAU,CAAV,
EAAa,KAAb,EAAoB,gBAApB,C;;MAEX,YAAY,KAAiB,CAAX,KAAW,C;MAC7B,IAAI,SAAU,CAAV,MAAiB,CAAr
B,C;UAEI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;aAER,IAAI,SAAU,EAAV,MAAiB,CAArB,C;QA
CH,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;aACJ,IAAI,SAAU,GAAV,MAAkB,GAAtB,C;QA
CH,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAGX,IAAI,SAAQ,CAAR,UAAa,QAAjB,C;QACI
,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAEX,YAAY,KAAiB,CAAX,KAAW,C;MAC7B,IAAI,SAAU,CAAV,MAAiB,CAAr
C;UAEI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;aAER,IAAI,SAAU,EAAV,MAAiB,CAArB,C;QA
CI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAGX,IAAI,SAAQ,CAAR,UAAa,QAAjB,C;QACI,OAAO,UAAU,CAAV,
EAAa,KAAb,EAAoB,gBAApB,C;;MAEX,YAAY,KAAiB,CAAX,KAAW,C;MAC7B,IAAI,SAAU,CAAV,MAAiB,CAAr
B,C;UAEI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;aAER,IAAI,SAAU,EAAV,MAAiB,CAArB,C;QA
CI,OAAO,UAAU,CAAV,

EAAsB,IAAtB,EAA4B,SAA5B,EAAuC,QAAvC,EAAiD,gBAAjD,C;UACX,IAAI,UAAQ,CAAZ,C;YACI,aAAc,gBAAO,gBAAP,C;YACd,yBAAa,CAAC,MAAD,IAAb,K;;YAEA,aAAc,gBAAY,OAAL,MAAK,CAAZ,C;YACd,wBAAa,CAAb,I;;eAGR,YAAS,CAAT,KAAc,EAAd,C;UACI,aAAW,eAAe,KAAf,EAAsB,IAAtB,EAA4B,SAA5B,EAAuC,QAAvC,EAAiD,gBAAjD,C;UACX,IAAI,UAAQ,CAAZ,C;YACI,aAAc,gBAAO,gBAAP,C;YACd,yBAAa,CAAC,MAAD,IAAb,K;;YAEA,WAAY,MAAD,GAAQ,KAAR,IAAqB,EAArB,GAA2B,K;YACtC,UAAW,WAAI,SAAS,C;UACxB,IAAI,UAAQ,CAAZ,C;YACI,aAAc,gBAAO,gBAAP,C;YACd,yBAAa,CAAC,MAAD,IAAb,K;;;UAIJ,IAAJ,UAAU,CAAV,EAAa,SAAb,EAAwB,gBAAxB,C;UACA,aAAc,gBAAO,gBAAP,C;;;MAK1B,OAAO,aAAc,W;K;ICtQzB,uC;MAU2D,OAAO,wB,CAAQ,2BAAwB,mBAAS,SAAT,C;K;IAEnF,oC;MAKI,OAAQ,OAAO,OAAW,mBAAS,SAAT,C;K;IAGvC,6C;MAMI,IAAI,cAAS,SAAT,C;UAAoB,OAAO,K;MAC/B,OAAO,IAAI,SAAS,C;K;IAGxB,2C;MAMI,OAAO,IAAI,SAAS,C;K;IAGxB,6C;MAMI

AHN,C;MAIA,yDAGK,OAHL,C;K;;IAxBA,gD;MAAA,yB;MAAA,wC;K;;IAIA,iD;MAAA,yB;MAAA,yC;K;;IAIA,iD;MAAA,yB;MAAA,yC;K;;IAIA,4C;MAAA,yB;MAAA,oC;K;;IAIA,4C;MAAA,yB;MAAA,oC;K;;IAIA,0C;MAAA,yB;MAAA,kC;K;;IAIA,yC;MAAA,yB;MAAA,iC;K;;IA3BJ,+B;MAAA,4Q;K;;IAAA,oC;MAAA,a;aAAA,a;UAAA,6C;aAAA,c;UAAA,8C;aAAA,c;UAAA,8C;aAAA,S;UAAA,yC;aAAA,S;UAAA,yC;aAAA,O;UAAA,uC;aAAA,M;UAAA,sC;;UAAA,6D;;K;;IAiCA,4D;MAGW,Q;MADP,0BAA2C,iBAAjB,UAAW,cAAM,EAAU,UAAW,cAAR,B,C;MAEvC,0BAAsB,CAAtB,C;QAA2B,gBAAS,UAAW,cAAX,GAAmB,UAAW,cAAvC,C;WAC3B,0BAAsB,CAAtB,C;QAA2B,gBAAS,UAAW,cAAX,GAAmB,UAAW,cAAvC,C;;QACnB,Y;MAHZ,W;K;IAOJ,oE;MAGW,Q;MADP,0BAA2C,iBAAjB,UAAW,cAAM,EAAU,UAAW,cAAR,B,C;MAEvC,0BAAsB,CAAtB,C;QAA2B,sBAA8C,uBAAW,cAAX,GAAmB,UAAW,cAAO,CAA9C,C;WAC3B,0BAAsB,CAAtB,C;QAA2B,iBAA8C,uBAAW,cAAX,GAAmB,UAAW,cAAO,CAA9C,C;;QACnB,Y;MAHZ,W;K;IAOJ,8D;MAGW,Q;MADP,0BAA2C,iBAAjB,UAAW,cAAM,EAAU,UAAW,cAAR,B,C;MAEvC,0BAAsB,CAAtB,C;QAAiC,YAAK,eAAW,cAAX,GAAmB,UAAW,cAAO,CAAtD

AqB,K;K;6EAuB1F,4B;MAEqE,iBAAY,KAAZ,C;K;6EAErE,mC;MAE4E,UAAY,KAAZ,IAAqB,K;K;6EAuBjG,4B;MAE+D,iBAAY,KAAZ,C;K;6EAE/D,mC;MAEsE,UAAY,KAAZ,IAAqB,K;K;6EAuB3F,4B;MAEgE,iBAAY,KAAZ,C;K;6EAEhE,mC;MAEuE,UAAY,KAAZ,IAAqB,K;K;6EAuB5F,4B;MAE6D,iBAAY,KAAZ,C;K;6EAE7D,mC;MAEoE,UAAY,KAAZ,IAAqB,K;K;6EAuBzF,4B;MAE8D,iBAAY,KAAZ,C;K;6EAE9D,mC;MAEqE,UAAY,KAAZ,IAAqB,K;K;6EAuB1F,4B;MAEiE,iBAAY,KAAZ,C;K;6EAEjE,mC;MAEwE,UAAY,KAAZ,IAAqB,K;K;6EAuB7F,4B;MAEkE,iBAAY,KAAZ,C;K;6EAElE,mC;MAEyE,UAAY,KAAZ,IAAqB,K;K;6GC3oC9F,wD;MAEqC,6B;QAAA,gBAA+B,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACpJ,QAAQ,E;MACR,EAAE,eAAF,IAAqB,a;MACrB,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;mIAiCX,+B;MAEgD,mC;QAAA,sBAAoC,K;MAAO,5E,QAAQ,E;MACR,EAAE,qBAAF,IAA2B,mB;MAC3B,OAAO,C;K;4EC9CX,4B;MAEgE,iBAAY,KAAZ,C;K;4EAgChE,4B;MAEyE,iBAAY,KAAZ,C;K;4EAiBzE,4B;MAEmE,iBAAY,KAAZ,C;K;4EAyBnE,4B;MAE0E,iBAAY,KAAZ,C;K;oIC7a1E,4H;MAE8C,qB;QAAA,QAAiB,E;MAAI,I,6B;QAAA,QAAgB,gBAAgC,C,E;MAAE,W,iC;QAAA,QAAoB,oBAA2D,2D,E;MAAE,W,iC;QAAA,QAAoB,oBAA2D,2D,E;MAAE,W,qC;QAAA,QAAoB,wBAAmE,mJ,U;;MAAnJ,qO,+B;QAAA,QAAoB,k

AAqB,a;MACrB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;wGAgDX,kQ;MAEoC,uB;QAAA,UAAoB,K;MAAO,wB;QAAA,WAAqB,K;MAAO,sB;QAAA,SAAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,0B;QAAA,aAAuB,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,+B;QAAA,kBAA4B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,oB;QAAA,OAAiB,I;MAAM,sB;QAAA,SAAe,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC7lB,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,YAAF,IAAkB,U;MAClB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,eAAF,IAAqB,a;MACrB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,eAAF,IAAqB,a;MACrB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;kGAsCX,iX;MAEiC,sB;QAAA,SAAkB,G;MAAK,sB;QAAA,SAAkB,G;MAAK,sB;QAAA,SAAkB,G;MAAK,yB;QAAA,YAAqB,C;MAAG,uB;QAAA,UAAoB,G;MAAK,uB;QAAA,UAAoB,G;MAAK,uB;QAAA,UAAoB,G;MAAK,uB;QAAA,UAAoB,G;MAAK,sB;QAAA,SAAmB,G;MAAK,sB;QAAA,SAAmB,C;MAAG,uB;QAAA,UAAoB,G;MAAK,wB;QAAA,WAAsB,C;MAAG,6B;QAAA,gBAA6B,I;MAAM,sB;QAAA,SAAkB,I;MAAM,uB;QAAA,UAAoB,K;MAAO,wB;QAAA,WAAqB,K;MAAO,sB;QAAA,SAAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,0B;QAAA,aAAuB,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,+B;QAAA,kBAA4B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,oB;QAAA,OAAiB,I;MAAM,sB;QAAA,SAAe,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAAO,sC;QAAA,yBAAsC,E;MAAC,QAAQ,E;MACr,EAAE,QAAF,IAAc,M;MACd,EAAE,QAAF,IAAc,M;MACd,EAAE,QAAF,IAAc,M;MACd,EAAE,QAAF,IAAc,M;MACd,EAAE,WAAF,IAAiB,S;MACjB,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,eAAF,IAAqB,a;MACrB,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,YAAF,IAAkB,U;MAClB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,eAAF,IAAqB,a;MACrB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,eAAF,IAAqB,a;MACrB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;kGA2C

Ac,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;8GAuBX,6D;MAEuC,oB;QAAA,OAAgB,E;MAAI,oB;QAAA,OAAgB,I;MAAM,sB;QAAA,SAAe,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC7K,QAAQ,E;MACR,EAAE,MAAF,IAAY,I;MACZ,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;wECnbX,4B;MAEyE,iBAAY,KAAZ,C;K;wEAEzE,2B;MAEgG,iBAAY,IAAZ,C;K;wEAwBhG,oC;MAE+F,UAAY,KAAZ,IAAqB,M;K;wEAmFpH,2B;MAEqE,iBAAY,IAAZ,C;K;wEAErE,kC;MAE2E,UAAY,IAAZ,IAAoB,K;K;wEAssC/F,4B;MAEyE,iBAAY,KAAZ,C;K;wEA0BzE,4B;MAEyE,iBAAY,KAAZ,C;K;wEAsBzE,4B;MAEuE,iBAAY,KAAZ,C;K;wEAyBvE,4B;MAE6E,iBAAY,KAAZ,C;K;2FA4C7E,gD;MAEiC,qB;QAAA,QAAiD,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAClK,QAAQ,E;MACR,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;uEAA+X,4B;MAEuE,iBAAY,KAAZ,C;K;wEAEvE,2B;MAE6F,iBAAY,IAAZ,C;K;wEAqN7F,4B;MAEyE,iBAAY,KAAZ,C;K;wEAEzE,2B;MAEqE,UAAY,KAAZ,IAAqB,M;K;+FAuehH,wD;MAEmC,6B;QAAA,gBAA8B,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACjK,QAAQ,E;MACR,EAAE,eAAF,IAAqB,a;MACrB,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;uGAuIX,mB;MAEuC,uB;QAAA,UAAoB,K;MAAO,QAAQ,E;MACvD,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,OAAF,IAAa,C;MACb,EAAE,K;+HAAyX,iB;MAEmC,qB;QAAA,QAAiD,I;MAAM,QAAQ,E;MACjK,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,OAAF,IAAa,C;MACb,EAAE,K;+FAAyX,sE;MAEiC,oB;QAAA,OAAgB,I;MAAM,wB;QAAA,WAAqB,K;MAAO,WA0+G4B,S;;MA1+GnB

WAAF,IAAiB,S;MACjB,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;2FAoFX,kF;MAEiC,uB;QAAA,UAAmB,E;MAAI,wB;QAAA,WAAoB,E;MAAI,sB;QAAA,SAAe,C;MAAG,qB;QAAA,QAAc,C;MAAG,qB;QAAA,QAAc,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAAjN,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,QAAF,IAAc,M;MACd,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;iHAyBX,0D;MAEqE,sB;QAAA,SAAe,S;MAAW,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAAzM,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;wEAmXX,4B;MAEkE,kB;QAAA,iBAAY,K;MAAO,oB;QAAA,mBAAc,K;MAAO,wB;QAAA,uBAAkB,K;MAAO,0B;QAAA,yBAAoB,K;MAAO,4B;QAAA,2BAAsB,K;MAAO,8B;QAAA,6BAAwB,K;MAAO,gC;QAAA,+BAA0B,K;MAAO,kC;QAAA,iCAA4B,K;MAAO,oC;QAAA,mCAA8B,K;MAAO,sC;QAAA,qCAAgC,K;MAAO,wC;QAAA,uCAAkC,K;MAAO,0C;QAAA,yCAAoC,K;MAAO,4C;QAAA,2CAAsC,K;MAAO,8C;QAAA,6CAAwC,K;MAAO,gD;QAAA,+CAA0C,K;MAAO,kD;QAAA,iDAA4C,K;MAAO,oD;QAAA,mDAA8C,K;MAAO,sD;QAAA,qDAAgD,K;MAAO,wD;QAAA,uDAAkD,K;MAAO,0D;QAAA,yDAAoD,K;MAAO,4D;QAAA,2DAAsD,K;MAAO,8D;QAAA,6DAAwD,K;MAAO,gE;QAAA,+DAA0D,K;MAAO,kE;QAAA,iEAA4D,K;MAAO,oE;QAAA,mEAA8D,K;MAAO,sE;QAAA,qEAAgE,K;MAAO,wE;QAAA,uEAAkE,K;MAAO,0E;QAAA,yEAAoE,K;MAAO,4E;QAAA,2EAAsE,K;MAAO,8E;QAAA,6EAAwE,K;MAAO,gF;QAAA,+EAA0E,K;MAAO,kF;QAAA,iFAA4E,K;MAAO,oF;QAAA,mFAA8E,K;MAAO,sF;QAAA,qFAAgF,K;MAAO,wF;QAAA,uFAAkF,K;MAAO,0F;QAAA,yFAAoF,K;MAAO,4F;QAAA,2FAAsF,K;MAAO,8F;QAAA,6FAAwF,K;MAAO,gG;QAAA,+FAA0F,K;MAAO,kG;QAAA,iGAA4F,K;MAAO,oG;QAAA,mGAA8F,K;MAAO,sG;QAAA,qGAAgG,K;MAAO,wG;QAAA,uGAAkG,K;MAAO,0G;QAAA,yGAAoG,K;MAAO,4G;QAAA,2GAAsG,K;MAAO,8G;QAAA,6GAAwG,K;MAAO,gH;QAAA,+GAA0G,K;MAAO,kH;QAAA,iHAA4G,K;MAAO,oH;QAAA,mHAA8G,K;MAAO,sH;QAAA,qHAAgH,K;MAAO,wH;QAAA,uHAAkH,K;MAAO,0H;QAAA,yHAAoH,K;MAAO,4H;QAAA,2HAAsH,K;MAAO,8H;QAAA,6HAAwH,K;MAAO,gI;QAAA,+HAA0H,K;MAAO,kI;QAAA,iIAA4H,K;MAAO,oI;QAAA,mIAA8H,K;MAAO,sI;QAAA,qIAAgI,K;MAAO,wI;QAAA,uIAAkI,K;MAAO,0I;QAAA,yIAAoI,K;MAAO,4I;QAAA,2IAAsI,K;MAAO,8I;QAAA,6IAAwI,K;MAAO,gJ;QAAA,+IAA0I,K;MAAO,kJ;QAAA,iJAA4I,K;MAAO,oJ;QAAA,mJAA8I,K;MAAO,sJ;QAAA,qJAAgJ,K;MAAO,wJ;QAAA,uJAAkJ,K;MAAO,0J;QAAA,yJAAoJ,K;MAAO,4J;QAAA,2JAAsJ,K;MAAO,8J;QAAA,6JAAwJ,K;MAAO,gK;QAAA,+JAA0J,K;MAAO,kK;QAAA,iKAA4J,K;MAAO,oK;QAAA,mKAA8J,K;MAAO,sK;QAAA,qKAAgK,K;MAAO,wK;QAAA,uKAAkK,K;MAAO,0K;QAAA,yKAAoK,K;MAAO,4K;QAAA,2KAAsK,K;MAAO,8K;QAAA,6KAAwK,K;MAAO,gL;QAAA,+KAA0K,K;MAAO,kL;QAAA,iLAA4K,K;MAAO,oL;QAAA,mLAA8K,K;MAAO,sL;QAAA,qLAAgL,K;MAAO,wL;QAAA,uLAAkL,K;MAAO,0L;QAAA,yLAAoL,K;MAAO,4L;QAAA,2LAAsL,K;MAAO,8L;QAAA,6LAAwL,K;MAAO,gM;QAAA,+LAA0L,K;MAAO,kM;QAAA,iMAA4L,K;MAAO,oM;QAAA,mMAA8L,K;MAAO,sM;QAAA,qMAAgM,K;MAAO,wM;QAAA,uMAAkM,K;MAAO,0M;QAAA,yMAAoM,K;MAAO,4M;QAAA,2MAAsM,K;MAAO,8M;QAAA,6MAAwM,K;MAAO,gN;QAAA,+MAA0M,K;MAAO,kN;QAAA,iNAA4M,K;MAAO,oN;QAAA,mNAA8M,K;MAAO,sN;QAAA,qNAAgN,K;MAAO,wN;QAAA,uNAAkN,K;MAAO,0N;QAAA,yNAAoN,K;MAAO,4N;QAAA,2NAAsN,K;MAAO,8N;QAAA,6NAAwN,K;MAAO,gO;QAAA,+NAA0N,K;MAAO,kO;QAAA,iOAA4N,K;MAAO,oO;QAAA,mOAA8N,K;MAAO,sO;QAAA,qOAAgO,K;MAAO,wO;QAAA,uOAAkO,K;MAAO,0O;QAAA,yOAAoO,K;MAAO,4O;QAAA,2OAAsO,K;MAAO,8O;QAAA,6OAAwO,K;MAAO,gP;QAAA,+OAA0O,K;MAAO,kP;QAAA,iPAA4O,K;MAAO,oP;QAAA,mPAA8O,K;MAAO,sP;QAAA,qPAAgP,K;MAAO,wP;QAAA,uPAAkP,K;MAAO,0P;QAAA,yPAAoP,K;MAAO,4P;QAAA,2PAAsP,K;MAAO,8P;QAAA,6PAAwP,K;MAAO,gQ;QAAA,+PAA0P,K;MAAO,kQ;QAAA,iQAA4P,K;MAAO,oQ;QAAA,mQAA8P,K;MAAO,sQ;QAAA,qQAAgQ,K;MAAO,wQ;QAAA,uQAAkQ,K;MAAO,0Q;QAAA,yQAAoQ,K;MAAO,4Q;QAAA,2QAAsQ,K;MAAO,8Q;QAAA,6QAAwQ,K;MAAO,gR;QAAA,+QAA0Q,K;MAAO,kR;QAAA,iRAA4Q,K;MAAO,oR;QAAA,mRAA8Q,K;MAAO,sR;QAAA,qRAAgR,K;MAAO,wR;QAAA,uRAAkR,K;MAAO,0R;QAAA,yRAAoR,K;MAAO,4R;QAAA,2RAAsR,K;MAAO,8R;QAAA,6RAAwR,K;MAAO,gS;QAAA,+RAA0R,K;MAAO,kS;QAAA,iSAA4R,K;MAAO,oS;QAAA,mSAA8R,K;MAAO,sS;QAAA,qSAAgS,K;MAAO,wS;QAAA,uSAAkS,K;MAAO,0S;QAAA,ySAAoS,K;MAAO,4S;QAAA,2SAAsS,K;MAAO,8S;QAAA,6SAAwS,K;MAAO,gT;QAAA,+SAA0S,K;MAAO,kT;QAAA,iTAA4S,K;MAAO,oT;QAAA,mTAA8S,K;MAAO,sT;QAAA,qTAAgT,K;MAAO,wT;QAAA,uTAAkT,K;MAAO,0T;QAAA,yTAAoT,K;MAAO,4T;QAAA,2TAAsT,K;MAAO,8T;QAAA,6TAAwT,K;MAAO,gU;QAAA,+TAA0T,K;MAAO,kU;QAAA,iUAA4T,K;MAAO,oU;QAAA,mUAA8T,K;MAAO,sU;QAAA,qUAAgU,K;MAAO,wU;QAAA,uUAAkU,K;MAAO,0U;QAAA,yUAAoU,K;MAAO,4U;QAAA,2UAAsU,K;MAAO,8U;QAAA,6UAAwU,K;MAAO,gV;QAAA,+UAA0U,K;MAAO,kV;QAAA,iVAA4U,K;MAAO,oV;QAAA,mVAA8U,K;MAAO,sV;QAAA,qVAAgV,K;MAAO,wV;QAAA,uVAAkV,K;MAAO,0V;QAAA,yVAAoV,K;MAAO,4V;QAAA,2VAAsV,K;MAAO,8V;QAAA,6VAAwV,K;MAAO,gW;QAAA,+VAA0V,K;MAAO,kW;QAAA,iWAA4V,K;MAAO,oW;QAAA,mWAA8V,K;MAAO,sW;QAAA,qWAAgW,K;MAAO,wW;QAAA,uWAAkW,K;MAAO,0W;QAAA,yWAAoW,K;MAAO,4W;QAAA,2WAAsW,K;MAAO,8W;QAAA,6WAAwW,K;MAAO,gX;QAAA,+WAA0W,K;MAAO,kX;QAAA,iXAA4W,K;MAAO,oX;QAAA,mXAA8W,K;MAAO,sX;QAAA,qXAAgX,K;MAAO,wX;QAAA,uXAAkX,K;MAAO,0X;QAAA,yXAAoX,K;MAAO,4X;QAAA,2XAAsX,K;MAAO,8X;QAAA,6XAAwX,K;MAAO,gY;QAAA,+XAA0X,K;MAAO,kY;QAAA,iYAA4X,K;MAAO,oY;QAAA,mYAA8X,K;MAAO,sY;QAAA,qYAAgY,K;MAAO,wY;QAAA,uYAAkY,K;MAAO,0Y;QAAA,yYAAoY,K;MAAO,4Y;QAAA,2YAAsY,K;MAAO,8Y;QAAA,6YAAwY,K;MAAO,gZ;QAAA,+YAA0Y,K;MAAO,kZ;QAAA,iZAA4Y,K;MAAO,oZ;QAAA,mZAA8Y,K;MAAO,sZ;QAAA,qZAAgZ,K;MAAO,wZ;QAAA,uZAAkZ,K;MAAO,0Z;QAAA,yZAAoZ,K;MAAO,4Z;QAAA,2ZAAsZ,K;MAAO,8Z;QAAA,6ZAAwZ,K;MAAO,ga;QAAA,+ZA0Z,K;MAAO,ka;QAAA,iaA4a,K;MAAO,oa;QAAA,maA8a,K;MAAO,sa;QAAA,qaAgb,K;MAAO,wa;QAAA,uaAkb,K;MAAO,0a;QAAA,yaAob,K;MAAO,4a;QAAA,2asb,K;MAAO,8a;QAAA,6awb,K;MAAO,gb;QAAA,+a0b,K;MAAO,kb;QAAA,ibA4b,K;MAAO,ob;QAAA,mbA8b,K;MAAO,sb;QAAA,qbAgc,K;MAAO,wb;QAAA,ubAkc,K;MAAO,0b;QAAA,ybAoc,K;MAAO,4b;QAAA,2bAsc,K;MAAO,8b;QAAA,6bAwc,K;MAAO,gc;QAAA,+b0c,K;MAAO,kc;QAAA,icA4c,K;MAAO,oc;QAAA,mcA8c,K;MAAO,sc;QAAA,qcAgd,K;MAAO,wc;QAAA,ucAkd,K;MAAO,0c;QAAA,ycAod,K;MAAO,4c;QAAA,2csd,K;MAAO,8c;QAAA,6cwd,K;MAAO,gd;QAAA,+d0d,K;MAAO,kd;QAAA,idA4d,K;MAAO,od;QAAA,mdA8d,K;MAAO,sd;QAAA,qdAge,K;MAAO,wd;QAAA,udAke,K;MAAO,0d;QAAA,ydAoe,K;MAAO,4d;QAAA,2dse,K;MAAO,8d;QAAA,6dwe,K;MAAO,gf;QAAA,+e0e,K;MAAO,kf;QAAA,ifA4e,K;MAAO,of;QAAA,mfA8e,K;MAAO,sf;QAAA,qfAgf,K;MAAO,wf;QAAA,ufAkf,K;MAAO,0f;QAAA,yfAof,K;MAAO,4f;QAAA,2fsf,K;MAAO,8f;QAAA,6fwf,K;MAAO,gg;QAAA,+g0f,K;MAAO,kg;QAAA,igA4f,K;MAAO,og;QAAA,mgA8f,K;MAAO,sg;QAAA,qgAgg,K;MAAO,wg;QAAA,ugAkg,K;MAAO,0g;QAAA,ygAog,K;MAAO,4g;QAAA,2gsg,K;MAAO,8g;QAAA,6gwg,K;MAAO,gh;QAAA,+h0g,K;MAAO,kh;QAAA,ihA4g,K;MAAO,oh;QAAA,mhA8g,K;MAAO,sh;QAAA,qhAgh,K;MAAO,wh;QAAA,uhAkh,K;MAAO,0h;QAAA,yhAoh,K;MAAO,4h;QAAA,2hsh,K;MAAO,8h;QAAA,6hwh,K;MAAO,gi;QAAA,+i0h,K;MAAO,ki;QAAA,iiA4h,K;MAAO,oi;QAAA,miA8h,K;MAAO,si;QAAA,qiAgi,K;MAAO,wi;QAAA,uiAki,K;MAAO,0i;QAAA,yioi,K;MAAO,4i;QAAA,2isi,K;MAAO,8i;QAAA,6iwi,K;MAAO,gj;QAAA,+j0i,K;MAAO,kj;QAAA,ijA4i,K;MAAO,oj;QAAA,mjA8i,K;MAAO,sj;QAAA,qjAgj,K;MAAO,wj;QAAA,ujAkj,K;MAAO,0j;QAAA,yjAoj,K;MAAO,4j;QAAA,2jsj,K;MAAO,8j;QAAA,6jwj,K;MAAO,gk;QAAA,+k0j,K;MAAO,kk;QAAA,ikA4j,K;MAAO,ok;QAAA,mkA8j,K;MAAO,sk;QAAA,qkAgk,K;MAAO,wk;QAAA,ukAkk,K;MAAO,0k;QAAA,ykAok,K;MAAO,4k;QAAA,2ksk,K;MAAO,8k;QAAA,6kwk,K;MAAO,gl;QAAA,+l0k,K;MAAO,kl;QAAA,ilA4k,K;MAAO,ol;QAAA,mlA8k,K;MAAO,sl;QAAA,qlAgl,K;MAAO,wl;QAAA,ulAkl,K;MAAO,0l;QAAA,ylAol,K;MAAO,4l;QAAA,2lsl,K;MAAO,8l;QAAA,6lwl,K;MAAO,gm;QAAA,+m0l,K;MAAO,km;QAAA,imA4l,K;MAAO,om;QAAA,mmA8l,K;MAAO,sm;QAAA,qmAgm,K;MAAO,wm;QAAA,umAkm,K;MAAO,0m;QAAA,ymAom,K;MAAO,4m;QAAA,2msm,K;MAAO,8m;QAAA,6mwm,K;MAAO,gn;QAAA,+n0m,K;MAAO,kn;QAAA,inA4m,K;MAAO,on;QAAA,mnA8m,K;MAAO,sn;QAAA,qnAgn,K;MAAO,wn;QAAA,unAkn,K;MAAO,0n;QAAA,ynAon,K;MAAO,4n;QAAA,2nsn,K;MAAO,8n;QAAA,6nwn,K;MAAO,go;QAAA,+o0n,K;MAAO,ko;QAAA,ioA4n,K;MAAO,oo;QAAA,moA8n,K;MAAO,so;QAAA,qoAgo,K;MAAO,wo;QAAA,uoAko,K;MAAO,0o;QAAA,yooo,K;MAAO,4o;QAAA,2oso,K;MAAO,8o;QAAA,6owo,K;MAAO,gp;QAAA,+p0o,K;MAAO,kp;QAAA,ipA4o,K;MAAO,op;QAAA,mpA8o,K;MAAO,sp;QAAA,qpAgp,K;MAAO,wp;QAAA,upAkp,K;MAAO,0p;QAAA,ypAop,K;MAAO,4p;QAAA,2psp,K;MAAO,8p;QAAA,6pwp,K;MAAO,gq;QAAA,+q0p,K;MAAO,kq;QAAA,iqA4p,K;MAAO,oq;QAAA,mqA8p,K;MAAO,sq;QAAA,qqAgq,K;MAAO,wq;QAAA,uqAkq,K;MAAO,0q;QAAA,yqAoq,K;MAAO,4q;QAAA,2qsq,K;MAAO,8q;QAAA,6qwq,K;MAAO,gr;QAAA,+r0q,K;MAAO,kr;QAAA,irA4q,K;MAAO,or;QAAA,mrA8q,K;MAAO,sr;QAAA,qrAgr,K;MAAO,wr;QAAA,urAkr,K;MAAO,0r;QAAA,yrAor,K;MAAO,4r;QAAA,2rsr,K;MAAO,8r;QAAA,6rwr,K;MAAO,gs;QAAA,+s0r,K;MAAO,ks;QAAA,isA4r,K;MAAO,os;QAAA,msA8r,K;MAAO,ss;QAAA,qsAgs,K;MAAO,ws;QAAA,usAks,K;MAAO,0s;QAAA,ysAos,K;MAAO,4s;QAAA,2sss,K;MAAO,8s;QAAA,6sws,K;MAAO,gt;QAAA,+t0s,K;MAAO,kt;QAAA,itA4s,K;MAAO,ot;QAAA,mtA8s,K;MAAO,st;QAAA,qtAgt,K;MAAO,wt;QAAA,utAkt,K;MAAO,0t;QAAA,ytAot,K;MAAO,4t;QAAA,2tst,K;MAAO,8t;QAAA,6twt,K;MAAO,gu;QAAA,+u0t,K;MAAO,ku;QAAA,iuA4t,K;MAAO,ou;QAAA,muA8t,K;MAAO,su;QAAA,quAgu,K;MAAO,wu;QAAA,uuAku,K;MAAO,0u;QAAA,yuAou,K;MAAO,4u;QAAA,2usu,K;MAAO,8u;QAAA,6uwu,K;MAAO,gv;QAAA,+v0u,K;MAAO,kv;QAAA,ivA4u,K;MAAO,ov;QAAA,mvA8u,K;MAAO,sv;QAAA,qvAgv,K;MAAO,wv;QAAA,uvAkv,K;MAAO,0v;QAAA,yvAov,K;MAAO,4v;QAAA,2vsv,K;MAAO,8v;QAAA,6vwv,K;MAAO,gw;QAAA,+w0v,K;MAAO,kw;QAAA,iwA4v,K;MAAO,ow;QAAA,mwA8v,K;MAAO,sw;QAAA,qwAgw,K;MAAO,ww;QAAA,uwAkw,K;MAAO,0w;QAAA,ywAow,K;MAAO,4w;QAAA,2wsw,K;MAAO,8w;QAAA,6www,K;MAAO,gx;QAAA,+x0w,K;MAAO,kx;QAAA,ixA4w,K;MAAO,ox;QAAA,mxA8w,K;MAAO,sx;QAAA,qxAgx,K;MAAO,wx;QAAA,uxAkx,K;MAAO,0x;QAAA,yxAox,K;MAAO,4x;QAAA,2xsx,K;MAAO,8x;QAAA,6xwx,K;MAAO,gy;QAAA,+y0x,K;MAAO,ky;QAAA,iyA4x,K;MAAO,oy;QAAA,myA8x,K;MAAO,sy;QAAA,qyAgy,K;MAAO,wy;QAAA,uyAky,K;MAAO,0y;QAAA,yyAoy,K;MAAO,4y;QAAA,2ysy,K;MAAO,8y;QAAA,6ywy,K;MAAO,gz;QAAA,+z0y,K;MAAO,kz;QAAA,izA4y,K;MAAO,oz;QAAA,mzA8y,K;MAAO,sz;QAAA,qzAgz,K;MAAO,wz;QAAA,uzAkz,K;MAAO,0z;QAAA,yzAoz,K;MAAO,4z;QAAA,2zsz,K;MAAO,8z;QAAA,6zwz,K;MAAO,g0;QAAA,+00z,K;MAAO,k0;QAAA,i0A4z,K;MAAO,o0;QAAA,m0A8z,K;MAAO,s0;QAAA,q0Ag0,K;MAAO,w0;QAAA,u0Ak0,K;MAAO,00;QAAA,y0Ao0,K;MAAO,40;QAAA,20s0,K;MAAO,80;QAAA,60w0,K;MAAO,g1;QAAA,+10,K;MAAO,k1;QAAA,i1A41,K;MAAO,o1;QAAA,m1A81,K;MAAO,s1;QAAA,q1Ag1,K;MAAO,w1;QAAA,u1Ak1,K;MAAO,01;QAAA,y1Ao1,K;MAAO,41;QAAA,21s1,K;MAAO,81;QAAA,61w1,K;MAAO,g2;QAAA,+20,K;MAAO,k2;QAAA,i2A42,K;MAAO,o2;QAAA,m2A82,K;MAAO,s2;QAAA,q2Ag2,K;MAAO,w2;QAAA,u2Ak2,K;MAAO,02;QAAA,y2Ao2,K;MAAO,42;QAAA,22s2,K;MAAO,82;QAAA,62w2,K;MAAO,g3;QAAA,+30,K;MAAO,k3;QAAA,i3A43,K;MAAO,o3;QAAA,m3A83,K;MAAO,s3;QAAA,q3Ag3,K;MAAO,w3;QAAA,u3Ak3,K;MAAO,03;QAAA,y3Ao3,K;MAAO,43;QAAA,23s3,K;MAAO,83;QAAA,63w3,K;MAAO,g4;QAAA,+40,K;MAAO,k4;QAAA,i4A44,K;MAAO,o4;QAAA,m4A84,K;MAAO,s4;QAAA,q4Ag4,K;MAAO,w4;QAAA,u4Ak4,K;MAAO,04;QAAA,y4Ao4,K;MAAO,44;QAAA,24s4,K;MAAO,84;QAAA,64w4,K;MAAO,g5;QAAA,+50,K;MAAO,k5;QAAA,i5A45,K;MAAO,o5;QAAA,m5A85,K;MAAO,s5;QAAA,q5Ag5,K;MAAO,w5;QAAA,u5Ak5,K;MAAO,05;QAAA,y5Ao5,K;MAAO,45;QAAA,25s5,K;MAAO,85;QAAA,65w5,K;MAAO,g6;QAAA,+60,K;MAAO,k6;QAAA,i6A46,K;MAAO,o6;QAAA,m6A86,K;MAAO,s6;QAAA,q6Ag6,K;MAAO,w6;QAAA,u6Ak6,K;MAAO,06;QAAA,y6Ao6,K;MAAO,46;QAAA,26s6,K;MAAO,86;QAAA,66w6,K;MAAO,g7;QAAA,+70,K;MAAO,k7;QAAA,i7A47,K;MAAO,o7;QAAA,m7A87,K;MAAO,s7;QAAA,q7Ag7,K;MAAO,w7;QAAA,u7Ak7,K;MAAO,07;QAAA,y7Ao7,K;MAAO,47;QAAA,27s7,K;MAAO,87;QAAA,67w7,K;MAAO,g8;QAAA,+80,K;MAAO,k8;QAAA,i8A48,K;MAAO,o8;QAAA,m8A88,K;MAAO,s8;QAAA,q8Ag8,K;MAAO,w8;QAAA,u8Ak8,K;MAAO,08;QAAA,y8Ao8,K;MAAO,48;QAAA,28s8,K;MAAO,88;QAAA,68w8,K;MAAO,g9;QAAA,+90,K;MAAO,k9;QAAA,i9A49,K;MAAO,o9;QAAA,m9A89,K;MAAO,s9;QAAA,q9Ag9,K;MAAO,w9;QAAA,u9Ak9,K;MAAO,09;QAAA,y9Ao9,K;MAAO,49;QAAA,29s9,K;MAAO,89;QAAA,69w9,K;MAAO,g+;QAAA,++0,K;MAAO,k+;QAAA,i+A4+,K;MAAO,o+;QAAA,m+A8+,K;MAAO,s+;QAAA,q+Ag+,K;MAAO,w+;QAAA,u+Ak+,K;MAAO,0+;QAAA,y+Ao+,K;MAAO,4+;QAAA,2+s+,K;MAAO,8+;QAAA,6+w+,K;MAAO,gA;QAAA,+A0+,K;MAAO,kA;QAAA,iAA4+,K;MAAO,oA;QAAA,mAA8+,K;MAAO,sA;QAAA,qAAg/,K;MAAO,wA;QAAA,uAAk/,K;MAAO,0A;QAAA,yAAo/,K;MAAO,4A;QAAA,2As/,K;MAAO,8A;QAAA,6Aw/,K;MAAO,gB;QAAA,+B0/,K;MAAO,kB;QAAA,iBA4/,K;MAAO,oB;QAAA,mBA8/,K;MAAO,sB;QAAA,qBAggB,K;MAAO,wB;QAAA,uBAkgB,K;MAAO,0B;QAAA,yBAogB,K;MAAO,4B;QAAA,2BsgB,K;MAAO,8B;QAAA,6BwgB,K;MAAO,gC;QAAA,+C0gB,K;MAAO,kC;QAAA,iCA4gB,K;MAAO,oC;QAAA,mCA8gB,K;MAAO,sC;QAAA,qCAghB,K;MAAO,wC;QAAA,uCAkhB,K;MAAO,0C;QAAA,yCAohB,K;MAAO,4C;QAAA,2CshB,K;MAAO,8C;QAAA,6CwhB,K;MAAO,gD;QAAA,+D0hB,K;MAAO,kD;QAAA,iDA4hB,K;MAAO,oD;QAAA,mDA8hB,K;MAAO,sD;QAAA,qDAgiB,K;MAAO,wD;QAAA,uDAkiB,K;MAAO,0D;QAAA,yDAoiB,K;MAAO,4D;QAAA,2DsiB,K;MAAO,8D;QAAA,6DwiB,K;MAAO,gE;QAAA,+E0iB,K;MAAO,kE;QAAA,iEA4iB,K;MAAO,oE;QAAA,mEA8iB,K;MAAO,sE;QAAA,qEAgjB,K;MAAO,wE;QAAA,uEAkjB,K;MAAO,0E;QAAA,yEAojB,K;MAAO,4E;QAAA,2EsjB,K;MAAO,8E;QAAA,6EwjB,K;MAAO,gF;QAAA,+F0jB,K;MAAO,kF;QAAA,iFA4jB,K;MAAO,oF;QAAA,mFA8jB,K;MAAO,sF;QAAA,qFAgkB,K;MAAO,wF;QAAA,uFAkkB,K;MAAO,0F;QAAA,yFAokB,K;MAAO,4F;QAAA,2FskB,K;MAAO,8F;QAAA,6FwkB,K;MAAO,gG;QAAA,+G0kB,K;MAAO,kG;QAAA,iGA4kB,K;MAAO,oG;QAAA,mGA8kB,K;MAAO,sG;QAAA,qGAglB,K;MAAO,wG;QAAA,uGAklB,K;MAAO,0G;QAAA,yGAolB,K;MAAO,4G;QAAA,2GslB,K;MAAO,8G;QAAA,6GwlB,K;MAAO,gH;QAAA,+H0lB,K;MAAO,kH;QAAA,iHA4lB,K;MAAO,oH;QAAA,mHA8lB,K;MAAO,sH;QAAA,qHAgmB,K;MAAO,wH;QAAA,uHAkmB,K;MAAO,0H;QAAA,yHAomB,K;MAAO,4H;QAAA,2HsmB,K;MAAO,8H;QAAA,6HwmB,K;MAAO,gI;QAAA,+I0mB,K;MAAO,kI;QAAA,iIA4mB,K;MAAO,oI;QAAA,mIA8mB,K;MAAO,sI;QAAA,qIAgnB,K;MAAO,wI;QAAA,uIAknB,K;MAAO,0I;QAAA,yIAonB,K;MAAO,4I;QAAA,2IsnB,K;MAAO,8I;QAAA,6IwnB,K;MAAO,gJ;QAAA,+J0nB,K;MAAO,kJ;QAAA,iJA4nB,K;MAAO,oJ;QAAA,mJA8nB,K;MAAO,sJ;QAAA,qJAgoB,K;MAAO,wJ;QAAA,uJAkoB,K;MAAO,0J;QAAA,yJAooB,K;MAAO,4J;QAAA,2JsoB,K;MAAO,8J;QAAA,6JwoB,K;MAAO,gK;QAAA,+K0oB,K;MAAO,kK;QAAA,iKA4oB,K;MAAO,oK;QAAA,mKA8oB,K;MAAO,sK;QAAA,qKAgpB,K;MAAO,wK;QAAA,uKAkpB,K;MAAO,0K;QAAA,yKAopB,K;MAAO,4K;QAAA,2KspB,K;MAAO,8K;QAAA,6KwpB,K;MAAO,gL;QAAA,+L0pB,K;MAAO,kL;QAAA,iLA4pB,K;MAAO,oL;QAAA,mLA8pB,K;MAAO,sL;QAAA,qLAgqB,K;MAAO,wL;QAAA,uLAkqB,K;MAAO,0L;QAAA,yLAoqB,K;MAAO,4L;QAAA,2LsqB,K;MAAO,8L;QAAA,6LwqB,K;MAAO,gM;QAAA,+M0qB,K;MAAO,kM;QAAA,iMA4qB,K;MAAO,oM;QAAA,mMA8qB,K;MAAO,sM;QAAA,qMAgrB,K;MAAO,wM;QAAA,uMAkrB,K;MAAO,0M;QAAA,yMAorB,K;MAAO,4M;QAAA,2MsrB,K;MAAO,8M;QAAA,6MwrB,K;MAAO,gN;QAAA,+N0rB,K;MAAO,kN;QAAA,iNA4rB,K;MAAO,oN;QAAA,mNA8rB,K;MAAO,sN;QAAA,qNAgsB,K;MAAO,wN;QAAA,uNAksB,K;MAAO,0N;QAAA,yNAosB,K;MAAO,4N;QAAA,2NssB,K;MAAO,8N;QAAA,6NwsB,K;MAAO,gO;QAAA,+O0sB,K;MAAO,kO;QAAA,iOA4sB,K;MAAO,oO;QAAA,mOA8sB,K;MAAO,sO;QAAA,qOAgtB,K;MAAO,wO;QAAA,uOAktB,K;MAAO,0O;QAAA,yOAotB,K;MAAO,4O;QAAA,2OstB,K;MAAO,8O;QAAA,6OwtB,K;MAAO,gP;QAAA,+P0tB,K;MAAO,kP;QAAA,iPA4tB,K;MAAO,oP;QAAA,mPA8tB,K;MAAO,sP;QAAA,qPAguB,K;MAAO,wP;QAAA,uPAkuB,K;MAAO,0P;QAAA,yPAouB,K;MAAO,4P;QAAA,2PsuB,K;MAAO,8P;QAAA,6PwuB,K;MAAO,gQ;QAAA,+Q0uB,K;MAAO,kQ;QAAA,iQA4uB,K;MAAO,oQ;QAAA,mQA8uB,K;MAAO,sQ;QAAA,qQAgvB,K;MAAO,wQ;QAAA,uQAkvB,K;MAAO,0Q;QAAA,yQAovB,K;MAAO,4Q;QAAA,2QsvB,K;MAAO,8Q;QAAA,6QwvB,K;MAAO,gR;QAAA,+R0vB,K;MAAO,kR;QAAA,iRA4vB,K;MAAO,oR;QAAA,mRA8vB,K;MAAO,sR;QAAA,qRAgwB,K;MAAO,wR;QAAA,uRAkwB,K;MAAO,0R;QAAA,yRAowB,K;MAAO,4R;QAAA,2RswB,K;MAAO,8R;QAAA,6RwwB,K;MAAO,gS;QAAA,+S0wB,K;MAAO,kS;QAAA,iSA4wB,K;MAAO,oS;QAAA,mSA8wB,K;MAAO,sS;QAAA,qSAgxB,K;MAAO,wS;QAAA,uSAkxB,K;MAAO,0S;QAAA,ySAoxB,K;MAAO,4S;QAAA,2SsxB,K;MAAO,8S;QAAA,6SwxB,K;MAAO,gT;QAAA,+T0xB,K;MAAO,kT;QAAA,iTA4xB,K;MAAO,oT;QAAA,mTA8xB,K;MAAO,sT;QAAA,qTAgyB,K;MAAO,wT;QAAA,uTAkyB,K;MAAO,0T;QAAA,yToyB,K;MAAO,4T;QAAA,2TsyB,K;MAAO,8T;QAAA,6TwyB,K;MAAO,gU;QAAA,+U0yB,K;MAAO,kU;QAAA,iUA4yB,K;MAAO,oU;QAAA,mUA8yB,K;MAAO,sU;QAAA,qUAgzB,K;MAAO,wU;QAAA,uUAkzB,K;MAAO,0U;QAAA,yUAozB,K;MAAO,4U;QAAA,2UszB,K;MAAO,8U;QAAA,6UwzB,K;MAAO,gV;QAAA,+V0zB,K;MAAO,kV;QAAA,iVA4zB,K;MAAO,oV;QAAA,mVA8zB,K;MAAO,sV;QAAA,qVAg0B,K;MAAO,wV;QAAA,uVAk0B,K;MAAO,0V;QAAA,yVAo0B,K;MAAO,4V;QAAA,2Vs0B,K;MAAO,8V;QAAA,6Vw0B,K;MAAO,gW;QAAA,+W0,K;MAAO,kW;QAAA,iWA40B,K;MAAO,oW;QAAA,mWA80B,K;MAAO,sW;QAAA,qWAg1B,K;MAAO,wW;QAAA,uWAk1B,K;MAAO,0W;QAAA,yWAo1B,K;MAAO,4W;QAAA,2Ws1B,K;MAAO,8W;QAAA,6Ww1B,K;MAAO,gX;QAAA,+X0,K;MAAO,kX;QAAA,iXA41B,K;MAAO,oX;QAAA,mXA81B,K;MAAO,sX;QAAA,qXAg2B,K;MAAO,wX;QAAA,uXAk2B,K;MAAO,0X;QAAA,yXAo2B,K;MAAO,4X;QAAA,2Xs2B,K;MAAO,8X;QAAA,6Xw2B,K;MAAO,gY;QAAA,+Y0,K;MAAO,kY;QAAA,iYA42B,K;MAAO,oY;QAAA,mYA82B,K;MAAO,sY;QAAA,qYAg3B,K;MAAO,wY;QAAA,uYAk3B,K;MAAO,0Y;QAAA,yYAo3B,K;MAAO,4Y;QAAA,2Ys3B,K;MAAO,8Y;QAAA,6Yw3B,K;MAAO,gZ;QAAA,+Z0,K;MAAO,kZ;QAAA,iZA43B,K;MAAO,oZ;QAAA,mZA83B,K;MAAO,sZ;QAAA,qZAg4B,K;MAAO,wZ;QAAA,uZAk4B,K;MAAO,0Z;QAAA,yZAo4B,K;MAAO,4Z;QAAA,2Zs4B,K;MAAO,8Z;QAAA,6Zw4B,K;MAAO,ga;QAAA,+a0,K;MAAO,ka;QAAA,iaA45B,K;MAAO,oa;QAAA,maA85B,K;MAAO,sa;QAAA,qaAg6B,K;MAAO,wa;QAAA,uaAk6B,K;MAAO,0a;QAAA,yaAo6B,K;MAAO,4a;QAAA,2as6B,K;MAAO,8a;QAAA,6aw6B,K;MAAO,gb;QAAA,+a0,K;MAAO,kb;QAAA,ibA46B,K;MAAO,ob;QAAA,mbA86B,K;MAAO,sb;QAAA,qbAg7B,K;MAAO,wb;QAAA,ubAk7B,K;MAAO,0b;QAAA,ybAo7B,K;MAAO,4b;QAAA,2bs7B,K;MAAO,8b;QAAA,6bw7B,K;MAAO,gc;QAAA,+b0,K;MAAO,kc;QAAA,icA47B,K;MAAO,oc;QAAA,mcA87B,K;MAAO,sc;QAAA,qcAg8B,K;MAAO,wc;QAAA,ucAk8B,K;MAAO,0c;QAAA,ycAo8B,K;MAAO,4c;QAAA,2cs8B,K;MAAO,8c;QAAA,6cw8B,K;MAAO,gd;QAAA,+d0,K;MAAO,kd;QAAA,idA49B,K

I,QAAQ,E;MACR,EAAE,MAAF,IAAY,I;MACZ,OAAO,C;K;wEAgBX,4B;MAEiE,iBAAY,KAAZ,C;K;wEAEjE,oC;MAE4E,iBAAY,aAAZ,C;K;wEAuT5E,4B;MAEmE,iBAAY,KAAZ,C;K;uFA2CnE,sB;MAE+B,iB;QAAA,IAAa,G;MAAK,iB;QAAA,IAAa,G;MAAK,iB;QAAA,IAAa,G;MAAK,iB;QAAA,IAAa,G;MAC9F,QAAQ,E;MACR,EAAE,GAAF,IAAS,C;MACT,EAAE,GAAF,IAAS,C;MACT,EAAE,GAAF,IAAS,C;MACT,EAAE,GAAF,IAAS,C;MACT,OAAO,C;K;qFA0CX,+B;MAE8B,iB;QAAA,IAAa,G;MAAK,iB;QAAA,IAAa,G;MAAK,qB;QAAA,QAAiB,G;MAAK,sB;QAAA,SAAkB,G;MACtG,QAAQ,E;MACR,EAAE,GAAF,IAAS,C;MACT,EAAE,GAAF,IAAS,C;MACT,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,OAAO,C;K;wEAOX,4B;MAEmE,iBAAY,KAAZ,C;K;yFAiHnE,oB;MAEgC,wB;QAAA,WAAY,2B+C,M;;MAAM,2B3E,QAAQ,E;MACR,EAAE,UAAF,IAAgB,IAAgB,Q;MAChB,OAAO,C;K;6FAgBX,+B;MAEkC,kC,oB;MAEA,wB;QAAA,WAAY,C,M;MAAM,wB;QAAA,WAAY,S,M;MAAM,wB;QAAA,WAAY,S,M;MAAM,wB;QAAA,WAAY,S

AuB,e;MACvB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,SAAF,IAAe,O;MACf,EAAE,cAAF,IAAoB,Y;MACpB
,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,SAAF,IAAe,O;MACf,OAAO,C;K;wHAsDX,wM;MAEyC,qB;QAAA,Q
AAqB,S;MAAW,sB;QAAA,SAAsB,S;MAAW,2B;QAAA,cAA4B,S;MAAW,yB;QAAA,YAA0B,S;MAAW,0B;Q
AAA,aAA6B,S;MAAW,0B;QAAA,aAA6B,S;MAAW,sB;QAAA,SAAuB,S;MAAW,0B;QAAA,aAA0B,S;MAAW
,0B;QAAA,aAA0B,S;MAAW,gC;QAAA,mBAAoC,S;MAAW,+B;QAAA,kBAAmC,S;MAAW,gC;QAAA,mBAA
oC,S;MAAW,uB;QAAA,UAAwB,S;MAAW,4B;QAAA,eAA4B,S;MAAW,wB;QAAA,WAAoB,S;MAAW,uB;QA
AA,UAAmB,S;MACtnB,QAAQ,E;MACR,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,aA
AF,IAAmB,W;MACnB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,YAAF,IAAkB,U;MAClB,EAAE,YAAF,IAAkB,U
;MAClB,EAAE,QAAF,IAAc,M;MACd,EAAE,YAAF,IAAkB,U;MAClB,EAAE,YAAF,IAAkB,U;MAClB,EAAE,
kBAAF,IAAwB,gB;MACxB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,SAAF,I
AAe,O;MACf,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,SAAF,IAAe,O;MACf,OA
AO,C;K;sHAYX,kN;MAEwC,wB;QAAA,WAA4C,S;MAAW,qB;QAAA,QAAiB,S;MAAW,sB;QAAA,SAAkB,S;
MAAW,2B;QAAA,cAAuB,S;MAAW,yB;QAAA,YAAqB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aA
AsB,S;MAAW,sB;QAAA,SAAkB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,gC;QAA
A,mBAA4B,S;MAAW,+B;QAAA,kBAA2B,S;MAAW,gC;QAAA,mBAA4B,S;MAAW,uB;QAAA,UAAmB,S;M
AAW,4B;QAAA,eAAwB,S;MAAW,wB;QAAA,WAAoB,S;MAAW,uB;QAAA,UAAmB,S;MAC9lB,QAAQ,E;M
ACR,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,
IAAmB,W;MACnB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,YAAF,IAAkB,U;MAClB,EAAE,YAAF,IAAkB,U;M
AClB,EAAE,QAAF,IAAc,M;MACd,EAAE,YAAF,IAAkB,U;MAClB,EAAE,YAAF,IAAkB,U;MAClB,EAAE,kB
AAF,IAAwB,gB;MACxB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,SAAF,IA
Ae,O;MACf,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,SAAF,IAAe,O;MACf,OAA
O,C;K;0HAsDX,wM;MAE0C,qB;QAAA,QAAiB,S;MAAW,sB;QAAA,SAAkB,S;MAAW,2B;QAAA,cAAuB,S;M
AAW,yB;QAAA,YAAqB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,sB;QAAA,SAAk
B,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,gC;QAAA,mBAA4B,S;MAAW,+B;QAA
A,kBAA2B,S;MAAW,gC;QAAA,mBAA4B,S;MAAW,uB;QAAA,UAAmB,S;MAAW,4B;QAAA,eAAwB,S;MAA
W,wB;QAAA,WAAoB,S;MAAW,uB;QAAA,UAAmB,S;MACziB,QAAQ,E;MACR,EAAE,OAAF,IAAa,K;MACb
,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,IAAmB,W;MACnB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,YAAF,I
AAkB,U;MAClB,EAAE,YAAF,IAAkB,U;MAClB,EAAE,QAAF,IAAc,M;MACd,EAAE,YAAF,IAAkB,U;MAClB
,EAAE,YAAF,IAAkB,U;MAClB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,kB
AAF,IAAwB,gB;MACxB,EAAE,SAAF,IAAe,O;MACf,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,UAAF,IAAgB,Q;
MAChB,EAAE,SAAF,IAAe,O;MACf,OAAO,C;K;gHAyDX,wM;MAEqC,qB;QAAA,QAAiB,S;MAAW,sB;QAAA,
SAAkB,S;MAAW,2B;QAAA,cAAuB,S;MAAW,yB;QAAA,YAAqB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;
QAAA,aAAsB,S;MAAW,sB;QAAA,SAAkB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aAAsB,S;MAA
W,gC;QAAA,mBAA4B,S;MAAW,+B;QAAA,kBAA2B,S;MAAW,gC;QAAA,mBAA4B,S;MAAW,uB;QAAA,UAA
mB,S;MAAW,4B;QAAA,eAAwB,S;MAAW,wB;QAAA,WAAoB,S;MAAW,uB;QAAA,UAAmB,S;MACxhB,Q
AAQ,E;MACR,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,IAAmB,W;MACnB,EA
AE,WAAF,IAAiB,S;MACjB,EAAE,YAAF,IAAkB,U;MAClB,EAAE,YAAF,IAAkB,U;MAClB,EAAE,QAAF,IA
Ac,M;MACd,EAAE,YAAF,IAAkB,U;MAClB,EAAE,YAAF,IAAkB,U;MAClB,EAAE,kBAAF,IAAwB,gB;MACx
B,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,SAAF,IAAe,O;MACf,EAAE,cAA
F,IAAoB,Y;MACpB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,SAAF,IAAe,O;MACf,OAAO,C;K;8HAqBX,gD;M
AEsE,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAChJ,QAAQ,E;MAC
R,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q
;MAChB,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,OAAO,C;K;sIAoBX,gD;MAEgD,qB;QAAA,QAAiB,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACjK,QAAQ,E;MACR,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,OAAO,C;K;wHAwCX,wB;
MAEyC,qB;QAAA,QAAiB,K;MAAO,qB;QAAA,QAAiB,K;MAAO,2C;QAAA,cAAuB,K;MAAO,yB;QAAA,YAAqB,K;MAAO,0B;QAAA,aAAsB,K;MAAO,0B;QAAA,aAAsB,K;MAAO,sB;QAAA,SAAkB,K;MAAO,0B;QAAA,aAAsB,K;MAAO,0B;QAAA,aAAsB,K;MAAO,gC;QAAA,mBAA4B,K;MAAO,+B;QAAA,kBAA2B,K;MAAO,gC;QAAA,mBAA4B,K;MAAO,uB;QAAA,UAAmB,K;MAAO,4B;QAAA,eAAwB,K;MAAO,wB;QAAA,WAAoB,K;MAAO,uB;QAAA,UAAm

YX,kC;MAEuC,qB;QAAA,QAAiB,S;MAAW,qB;QAAA,QAAiB,S;MAAW,mB;QAAA,MAAe,S;MAAW,mB;Q
AAA,MAAe,S;MACpI,QAAQ,E;MACR,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MACb,EAAE,KAA
F,IAAW,G;MACX,EAAE,KAAF,IAAW,G;MACX,OAAO,C;K;gGAYX,oB;MAE6B,mB;QAAA,MAAY,S;MAA
W,mB;QAAA,MAAY,S;MAC5D,QAAQ,E;MACR,EAAE,KAAF,IAAW,G;MACX,EAAE,KAAF,IAAW,G;MAC
X,OAAO,C;K;kHAYX,kC;MAEsC,qB;QAAA,QAAc,S;MAAW,qB;QAAA,QAAc,S;MAAW,mB;QAAA,MAAY,
S;MAAW,mB;QAAA,MAAY,S;MACvH,QAAQ,E;MACR,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;
MACb,EAAE,KAAF,IAAW,G;MACX,EAAE,KAAF,IAAW,G;MACX,OAAO,C;K;gIAeX,wB;MAE6C,qB;QAA
A,QAAkB,S;MAAW,qB;QAAA,QAAkB,S;MACxF,QAAQ,E;MACR,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF
,IAAa,K;MACb,OAAO,C;K;oIAeX,wB;MAE+C,qB;QAAA,QAAiB,S;MAAW,qB;QAAA,QAAiB,S;MACxF,QA
AQ,E;MACR,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MACb,OAAO,C;K;4FAKX,Y;MAGI,QAAQ,E
;MACR,OAAO,C;K;oFAKX,Y;MAGI,QAAQ,E;MACR,OAAO,C;K;8FAKX,Y;MAGI,QAAQ,E;MACR,OAAO,C
;K;kGAUX,oB;MAE8B,wB;QAAA,WAAkC,S;MAC5D,QAAQ,E;MACR,EAAE,UAAF,IAAgB,Q;MAChB,OAA
O,C;K;4FAUmE,qB;MAAQ,OAAO,M;K;8FAAd,qB;MAAQ,OAAQ,O;K;4FASrB,qB;MAAQ,OAAO,M;K;0GAE
R,qB;MAAQ,OAAc,a;K;8FAAE7B,qB;MAAQ,OAAO,M;K;gGAAEd,qB;MAAQ,OAAQ,O;K;8FASjB,qB;MAAQ,O
AAO,M;K;gHAAEL,qB;MAAQ,OAAiB,gB;K;wGASrC,qB;MAAQ,OAAa,Y;K;0GAEpB,qB;MAAQ,OAAc,a;K;w
GAEvB,qB;MAAQ,OAAa,Y;K;oFCroB7F,4B;MAE6E,iBAAY,KAAZ,C;K;iGASnB,qB;MAAQ,OAAS,Q;K;6FAE
nB,qB;MAAQ,OAAO,M;K;+FAAd,qB;MAAQ,OAAQ,O;K;iGASF,qB;MAAQ,OAAU,S;K;+FAEnB,qB;MAAQ,O
AAS,Q;K;mGAS3B,qB;MAAQ,OAAW,U;K;mGAEnB,qB;MAAQ,OAAW,U;K;6GC1D/E,mb;MAEmC,yB;QAAA
A,YAAkB,C;MAAG,qB;QAAA,QAAiB,G;MAAK,sB;QAAA,SAAkB,G;MAAK,wB;QAAA,WAAmB,G;MAAI,k
C;QAAA,qBAA6B,G;MAAI,qB;QAAA,QAAc,C;MAAG,qB;QAAA,QAAc,C;MAAG,qB;QAAA,QAAc,C;MAA
G,2B;QAAA,cAAuB,E;MAAI,yB;QAAA,YAAsB,K;MAAO,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;
MAAG,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,sB;QAAA,SAAiB,C;MAAG,uB;QAAA,UA
AkB,C;MAAG,6B;QAAA,gBAA8B,I;MAAM,sB;QAAA,SAAkB,I;MAAM,uB;QAAA,UAAoB,K;MAAO,wB;QA
AA,WAAqB,K;MAAO,sB;QAAA,SAAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,gC;QAAA,mBAA6B,K;MA
AO,gC;QAAA,mBAA6B,K;MAAO,0B;QAAA,aAAuB,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,6B;QAAA,gBA
A0B,K;MAAO,+B;QAAA,kBAA4B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,8B;
QAAA,iBAA2B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,oB;QAAA,OAAgB,I;MAAM,sB;QAAA,SAAe,C;MA
AG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACl/B,QAAQ,E;MACR
,EAAE,WAAF,IAAiB,S;MACjB,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,C;MACd,EAAE,UAAF,IAAgB,Q
;MAChB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MACb,EA
AE,OAAF,IAAa,K;MACb,EAAE,aAAF,IAAmB,W;MACnB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,SAAF,IAAe,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,I
AAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,eAAF,IAAqB,a;MACrB,EAAE,QAAF,IAAc,C;MACd,EAAE
,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,QAAF,IAAc,C;MACd,EAAE,SAAF,IAAe,O;MA
Cf,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,YAAF,IAAkB,U;MAClB,EAA
E,gBAAF,IAAsB,c;MACtB,EAAE,eAAF,IAAqB,a;MACrB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,oBAAF,IAA
0B,kB;MAC1B,EAAE,eAAF,IAAqB,a;MACrB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,oBAAF,IAA0B,kB;MAC
1B,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,C;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,Ik
B,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;6GC1BX,0C;MAEwC,oB;QAAA,OAAiB,I;MAAM,sB
;QAAA,SAAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,uB;QAAA,UAAoB,K;MACpI,QAAQ,E;MACR,EAAE,
MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,C;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,
OAAO,C;K;4EAmIX,4B;MAEkE,iBAAY,KAAZ,C;K;4EAElE,qC;MAE2E,UAAY,KAAZ,IAAqB,O;K;4EAiBhG,
4B;MAEuE,iBAAY,KAAZ,C;K;4EAEvE,qC;MAE+E,UAAY,KAAZ,IAAqB,O;K;4EAiBpG,4B;MAEuE,iBAAY,
KAAZ,C;K;4EAEvE,qC;MAE+E,UAAY,KAAZ,IAAqB,O;K;4EAiGpG,4B;MAEoE,iBAAY,KAAZ,C;K;2EAEpE,
qC;MAE4E,UAAY,KAAZ,IAAqB,O;K;4EAkcjG,4B;MAE6E,iBAAY,KAAZ,C;K;4EAE7E,qC;MAEqF,UAAY,K
AAZ,IAAqB,O;K;4EAgP1G,4B;MAEqE,iBAAY,KAAZ,C;K;4EAErE,qC;MAE6E,UAAY,KAAZ,IAAqB,O;K;uFJ
57BlG,+H;MAE8B,sB;QAAA,SAAkB,S;MAAW,uB;QAAA,UAAmB,S;MAAW,oB;QAAA,OAAgB,S;MAAW,w
B;QAAA,WAAoB,S;MAAW,8B;QAAA,iBAA0B,S;MAAW,oB;QAAA,OAAqB,S;MAAW,2B;QAAA,cAAmC,S;

MAAW,qB;QAAA,QAAuB,S;MAAW,wB;QAAA,WAA6B,S;MAAW,yB;QAAA,YAAqB,S;MAAW,yB;QAAA,YAAsB,S;MAAW,wB;QAAA,WAAe,S;MAC5Z,QAAQ,E;MACR,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,MAAF,IAAY,I;MACZ,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,MAAF,IAAY,I;MACZ,EAAE,aAAF,IAAmB,W;MACnB,EAAE,OAAF,IAAa,K;MACb,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,QAAF,IAAc,Q;MACd,OAAO,C;K;yFA0CX,uC;MAE+B,sB;QAAA,SAAiB,G;MAAK,0B;QAAA,aAAsB,I;MAAM,uB;QAAA,UAAmB,S;MAChG,QAAQ,E;MACR,EAAE,QAAF,IAAc,M;MACd,EAAE,YAAF,IAAkB,U;MAClB,EAAE,SAAF,IAAe,O;MACf,OAAO,C;K;qFAUgD,qB;MAAQ,OAAG,E;K;mFAAX,qB;MAAQ,OAAQ,O;K;iFAAR,jB,qB;MAAQ,OAAO,M;K;mFAAf,qB;MAAQ,OAAQ,O;K;qFAAR,qB;MAAQ,OAAS,Q;K;mFAAT,lB,qB;MAAQ,OAAQ,O;K;mFAAR,hB,qB;MAAQ,OAAQ,O;K;mFAAR,hB,qB;MAAQ,OAAQ,O;K;qFAAR,qB;MAAQ,OAAG,E;K;yFAAH,qB;MAAQ,OAAW,U;K;mFAAX,tB,qB;MAAQ,OAAQ,O;K;mFAAR,jB,qB;MAAQ,OAAO,M;K;qFAAf,qB;MAAQ,OAAQ,O;K;yFAAR,eb,qB;MAAQ,OAAW,U;K;mFAAX,tB,qB;MAAQ,OAAQ,O;K;qFAAf,qB;MAAQ,OAAS,Q;K;qFAAjB,qB;MAAQ,OAAS,Q;K;uFAAjB,qB;MAAQ,OAAS,Q;K;mGAAV,qB;MAAQ,OAAG,e;K;iGAAH,qB;MAAQ,OAAK,c;K;qFAAb,9B,qB;MAAQ,OAAQ,O;K;qFAAR,qB;MAAQ,OAAS,Q;K;iFAAT,nB,qB;MAAQ,OAAO,M;K;yFAAf,qB;MAAQ,OAAW,U;K;+FAAX,hB,qB;MAAQ,OAAK,a;K;uFAAb,1B,qB;MAAQ,OAAO,S;K;iFAAP,rB,qB;MAAQ,OAAO,M;K;iFAAR,S,qB;MAAQ,OAAO,M;K;iGAAR,R,qB;MAAQ,OAAK,a;K;uFAAb,1B,qB;MAAQ,OAAO,S;K;yFAAP,9B,qB;MAAQ,OAAW,U;K;yFAAX,jB,qB;MAAQ,OAAW,U;K;qFAAX,rB,qB;MAAQ,OAAS,Q;K;yFAAT,f,qB;MAAQ,OAAW,U;K;+FAAX,hB,qB;MAAQ,OAAK,a;K;qGAAL,nB,qB;MAAQ,OAAO,S;K;iGAAP,nB,qB;MAAQ,OAAO,M;K;uFAAf,1B,qB;MAAQ,OAAS,Q;K;mFAAT,lB,qB;MAAQ,OAAQ,O;K;uFAAR,f,qB;MAAQ,OAAS,Q;K;mFAAT,xB,qB;MAAQ,OAAQ,O;K;mFAAR,jB,qB;MAAQ,OAAO,M;K;yFAAP,Z,qB;MAAQ,OAAO,S;K;qFAAP,pB,qB;MAAQ,OAAQ,O;K;qFAAR,f,qB;MAAQ,OAAS,Q;K;qGAAT,T,qB;MAAQ,OAAO,S;K;+FAAP,nB,qB;MAAQ,OAAK,a;K;qGAAL,nB,qB;MAAQ,OAAO,S;K;+FAAP,nB,qB;MAAQ,OAAK,a;K;qGAAL,nB,qB;MAAQ,OAAO,S;K;yFAAP,9B,qB;MAAQ,OAAW,U;K;yFAAX,jB,qB;MAAQ,OAAW,U;K;qFAAX,rB,qB;MAAQ,OAAS,Q;K;qFAAT,lB,qB;MAAQ,OAAS,Q;K;uFAAT

AO,C;K;yGAmBX,yC;MAEsC,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAChH,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;yGAsBX,2B;MAGI,QAAQ,E;MACR,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,OAAO,C;K;+FA8BX,sE;MAEoD,wB;QAAA,WAAoB,I;MAAM,wB;QAAA,WAAqB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACpL,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;6GAuBX,0D;MAAE2D,sB;QAAA,SAAkB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/J,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;2GAaX,qC;MAAE4D,sB;QAAA,SAAkB,M;MAAQ,sB;QAAA,SAAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/J,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;uHAAX,mF;MAAE6E,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAAmB,M;MAAQ,sB;QAAA,SAAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE8D,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAAmB,M;MAAQ,sB;QAAA,SAAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE+D,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAAmB,M;MAAQ,sB;QAAA,SAAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;uGAAX,iC;MAAEgE,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAAmB,M;MAAQ,sB;QAAA,SAAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEiE,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAAmB,M;MAAQ,sB;QAAA,SAAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEkE,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAAmB,M;MAAQ,sB;QAAA,SAAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEmE,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAAmB,M;MAAQ,sB;QAAA,SAAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEoE,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAAmB,M;MAAQ,sB;QAAA,SAAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEqE,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAAmB,M;MAAQ,sB;QAAA,SAAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEsE,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEuE,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEwE,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEyE,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE0E,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE2E,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE4E,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE6E,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE8E,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE+E,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEgF,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEiF,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEkF,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEmF,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEoF,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEqF,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEsF,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEuF,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEwF,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEyF,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE0F,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE2F,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE4F,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE6F,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE8F,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE+F,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEgG,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEiG,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEkG,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEmG,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEoG,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEqG,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEsG,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEuG,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEwG,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEyG,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE0G,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE2G,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE4G,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE6G,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE8G,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAE+G,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEgH,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEiH,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEkH,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MAClB,EAAE,UAAF,IAAgB,Q;MAChB,OAAO,C;K;qGAAX,+B;MAAEmH,oB;QAAA,OAAgB,S;MAAW,sB;QAAA,SAmB,M;MAAQ,sB;QAAA,SAmB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/L,QAAQ,E;MACR,EAAE,SAAF,IAAe,O;MACf,EAAE,

C;K;mDAE5C,iB;MAI4D,yBAAgB,IAAhB,EAAsB,KAAtB,C;K;;IC/BhE,8B;MAAA,e;MAAA,iB;MAAA,uB;K;IAAA,4B;MAAA,+B;O;MACI,4C;MACA,kD;MACA,0C;MACA,8C;K;;IAHA,mC;MAAA,kB;MAAA,2B;K;;IACA,sC;MAAA,kB;MAAA,8B;K;;IACA,kC;MAAA,kB;MAAA,0B;K;;IACA,oC;MAAA,kB;MAAA,4B;K;;IAJJ,wB;MAAA,sH;K;;IAAA,6B;MAAA,a;aAAA,O;UAAA,gC;aAAA,U;UAAA,mC;aAAA,M;UAAA,+B;aAAA,Q;UAAA,iC;;UAAA,6D;;K;;IAOA,4B;MAKI,mD;MACA,2BAA4B,I;K;yCAE5B,Y;MAEiB,IAAN,I;M5JUX,IAAI,E4JXQ,mD5JWR,CAAJ,C;QACI,cAAa,qB;QAAb,eAAM,MAAM,gCAAyB,OAAQ,WAAjC,C;;M4JZX,CAAC,oBAAN,M;aAAM,oBAAN,M;UAAM,oBAAN,M;aAAM,oBAAN,M;UAAM,oBAAN,M;aAAM,oBAAN,M;UAAM

OAAZ,C;K;IAI3C,sG;MAAA,kD;K;8FACH,Y;MAAkC,OAAA,0BAAc,U;K;2FAChD,Y;MAAyB,OAAA,0BAAc,OAAO,I;K;;wEAJtD,Y;MACI,oBAAoB,6BAAQ,W;MAC5B,+F;K;sHAMmB,Y;MAAQ,OAAA,qBAAiB,K;K;;mFAb5D,Y;MACI,IAAI,4BAAJ,C;QACI,+E;;MAcJ,OAAO,mC;K;IAOwD,uD;MAAA,qB;QAAE,2CAAS,EAAT,C;O;K;qCAAzE,Y;MAAkC,OAAQ,eAAR,YAAQ,EAAa,IAAb,EAAmB,GAAnB,EAAwB,GAAxB,kBAA6B,iCAA7B,C;K;+CAA1C,iB;MAAuD,+BAAS,KAAM,IAAf,IAAsB,GAAtB,wBAAS,KAAM,MAAf,C;K;+CAEnF,a;MAAwC,OAAI,MAAM,IAAV,GAAgB,YAAhB,GAA6B,SAA7B,EAAuC,C;K;IAWtD,4E;MAAA,wC;MAAS,6B;K;gFAAf,mB;MAAsE,iDAAc,OAAd,C;K;IAI3D,wG;MAAA,kD;K;gGACH,Y;MAAkC,OAAA,0BAAc,U;K;6FAChD,Y;MAAyB,OAAA,0BAAc,OAAO,M;K;;0EAJtD,Y;MACI,oBAAoB,6BAAQ,W;MAC5B,iG;K;wHAMmB,Y;MAAQ,OAAA,qBAAiB,K;K;;qFAb5D,Y;MACI,IAAI,8BAAJ,C;QACI,mF;;MAcJ,OAAO,qC;K;oDAAP,mB;MAAsE,gCAAS,KAAM,IAAf,IAAsB,GAAtB,wBAAS,KAAM,MAAf

BAAS,OAAT,C;QACA,M;;MAGJ,sBAAe,YAAO,CAAP,IAAf,C;MA2BA,oBAjIgD,mBAAY,cAiI1B,KAjI0B,IAAZ,C;MAmIhD,IAAI,QAAS,SAAD,GAAQ,CAAR,IAAe,CAA3B,C;QAEI,+BAA+B,mBAAY,aAAZ,C;QAC/B,sBAAsB,mBAAY,WAAZ,C;QAEtB,IAAI,4BAA4B,WAAhC,C;UACI,mBAAY,eAAZ,IAA+B,mBAAY,WAAZ,C;U/JgrB3C,U+J/qBY,kB/J+qBZ,E+J/qBiC,kB/J+qBjC,E+J/qB8C,W/J+qB9C,E+J/qBoD,cAAO,CAAP,I/J+qBpD,E+J/qB8D,2BAA2B,CAA3B,I/J+qB9D,C;;UAAA,U+J7qBY,kB/J6qBZ,E+J7qBiC,kB/J6qBjC,E+J7qB8C,cAAO,CAAP,I/J6qB9C,E+J7qBwD,W/J6qBxD,E+J7qB8D,kBAAY,O/J6qB1E,C;U+J5qBY,mBAAY,kBAAY,OAAZ,GAAmB,CAAnB,IAAwB,CAAxB,IAAZ,C;MAAoC,CAAC,I/J4qBzC,E+J5qB+C,2BAA2B,CAA3B,I/J4qB/C,C;U+J3qBY,mBAAY,CAAZ,C;U/J2qBZ,I+J3qBkB,kB/J2qBlB,E+J3qBuC,kB/J2qBvC,E+J3qB4C,2BAA2B,CAA3B,I/J2qB5C,C;U+J1qBY,mBAAY,CAAZ,C;U/J0qBZ,I+J1qBkB,kB/J0qBlB,E+J1qBuC,2BAA2B,CAA3B,I/J0qBvC,C;U+JzqBY,mBAAY,CAAZ,C;U/JyqBZ,I+JzqBkB,kB/JyqBlB,E+JzqBuC,2BAA2B,CAA3B,I/JyqBvC,C;U+JxqBY,mBAAY,CAAZ,C;U/JwqBZ,I+JxqBkB,kB/JwqBlB,E+JxqBuC,2BAA2B,CAA3B,I/JwqBvC,C;U+JvqBY,mBAAY,CAAZ,C;U/JuqBZ,I+JvqBkB,kB/JuqBlB,E+JvqBuC,2BAA2B,CAA3B,I/JuqBvC,C;U+JtqBY,mBAAY,CAAZ,C;U/JsqBZ,I+JtqBkB,kB/JsqBlB,E+JtqBuC,2BAA2B,CAA3B,I/JsqBvC,C;U+JrqBY,mBAAY,CAAZ,C;U/JqqBZ,I+JrqBkB,kB/JqqBlB,E+JrqBuC,2BAA2B,CAA3B,I/JqqBvC,C;U+JpqBY,mBAAY,CAAZ,C;U/JoqBZ,I+JpqBkB,kB/JoqBlB,E+JpqBuC,2BAA2B,CAA3B,I/JoqBvC,C;U+JnqBY,mBAAY,CAAZ,C;U/JmqBZ,I+JnqBkB,kB/JmqBlB,E+JnqBuC,2BAA2B,CAA3B,I/JmqBvC,C;U+JlqBY,mBAAY,CAAZ,C;U/JkqBZ,I+JlqBkB,kB/JkqBlB,E+JlqBuC,2BAA2B,CAA3B,I/JkqBvC,C;U+JjqBY,mBAAY,CAAZ,C;U/JiqBZ,I+JjqBkB,kB/JiqBlB,E+JjqBuC,2BAA2B,CAA3B,I/JiqBvC,C;U+JhqBY,mBAAY,CAAZ,C;U/JgqBZ,I+JhqBkB,kB/JgqBlB,E+JhqBuC,2BAA2B,CAA3B,I/JgqBvC,C;U+JfqBY,mBAAY,CAAZ,C;U/JeqBZ,I+JfqBkB,kB/JeqBlB,E+JfqBuC,2BAA2B,CAA3B,I/JeqBvC,C;U+JdqBY,mBAAY,CAAZ,C;U/JcqBZ,I+JdqBkB,kB/JcqBlB,E+JdqBuC,2BAA2B,CAA3B,I/JcqBvC,C;U+JbqBY,mBAAY,CAAZ,C;U/JaqBZ,I+JbqBkB,kB/JaqBlB,E+JbqBuC,2BAA2B,CAA3B,I/JaqBvC,C;U+JZqBY,mBAAY,CAAZ,C;U/JYqBZ,I+JZqBkB,kB/JYqBlB,E+JZqBuC,2BAA2B,CAA3B,I/JYqBvC,C;U+JXqBY,mBAAY,CAAZ,C;U/JWqBZ,I+JXqBkB,kB/JWqBlB,E+JXqBuC,2BAA2B,CAA3B,I/JWqBvC,C;U+JVqBY,mBAAY,CAAZ,C;U/JUqBZ,I+JVqBkB,kB/JUqBlB,E+JVqBuC,2BAA2B,CAA3B,I/JUqBvC,C;U+JTqBY,mBAAY,CAAZ,C;U/JSqBZ,I+JTqBkB,kB/JSqBlB,E+JTqBuC,2BAA2B,CAA3B,I/JSqBvC,C;U+JRqBY,mBAAY,CAAZ,C;U/JQqBZ,I+JRqBkB,kB/JQqBlB,E+JRqBuC,2BAA2B,CAA3B,I/JQqBvC,C;U+JPqBY,mBAAY,CAAZ,C;U/JOqBZ,I+JPqBkB,kB/JOqBlB,E+JPqBuC,2BAA2B,CAA3B,I/JOqBvC,C;U+JNqBY,mBAAY,CAAZ,C;U/JMqBZ,I+JNqBkB,kB/JMqBlB,E+JNqBuC,2BAA2B,CAA3B,I/JMqBvC,C;U+JLqBY,mBAAY,CAAZ,C;U/JKqBZ,I+JLqBkB,kB/JKqBlB,E+JLqBuC,2BAA2B,CAA3B,I/JKqBvC,C;U+JJqBY,mBAAY,CAAZ,C;U/JIqBZ,I+JJqBkB,kB/JIqBlB,E+JJqBuC,2BAA2B,CAA3B,I/JIqBvC,C;U+JHqBY,mBAAY,CAAZ,C;U/JGqBZ,I+JHqBkB,kB/JGqBlB,E+JHqBuC,2BAA2B,CAA3B,I/JGqBvC,C;U+JFqBY,mBAAY,CAAZ,C;U/JEqBZ,I+JFqBkB,kB/JEqBlB,E+JFqBuC,2BAA2B,CAA3B,I/JEqBvC,C;U+JDqBY,mBAAY,CAAZ,C;U/JCqBZ,I+JDqBkB,kB/JCqBlB,E+JDqBuC,2BAA2B,CAA3B,I/JCqBvC,C;U+JBqBY,mBAAY,CAAZ,C;U/JAqBZ,I+JBqBkB,kB/JAqBlB,E+JBqBuC,2BAA2B,CAA3B,I/JAqBvC,C;U+JCqBY,mBAAY,CAAZ,C;U/JDqBZ,I+JCqBkB,kB/JDqBlB,E+JCqBuC,2BAA2B,CAA3B,I/JDqBvC,C;U+JEqBY,mBAAY,CAAZ,C;U/JFqBZ,I+JEqBkB,kB/JFqBlB,E+JEqBuC,2BAA2B,CAA3B,I/JFqBvC,C;U+JGqBY,mBAAY,CAAZ,C;U/JHqBZ,I+JGqBkB,kB/JHqBlB,E+JGqBuC,2BAA2B,CAA3B,I/JHqBvC,C;U+JIqBY,mBAAY,CAAZ,C;U/JJqBZ,I+JIqBkB,kB/JJqBlB,E+JIqBuC,2BAA2B,CAA3B,I/JJqBvC,C;U+JKqBY,mBAAY,CAAZ,C;U/JLqBZ,I+JKqBkB,kB/JLqBlB,E+JKqBuC,2BAA2B,CAA3B,I/JLqBvC,C;U+JMqBY,mBAAY,CAAZ,C;U/JNqBZ,I+JMqBkB,kB/JNqBlB,E+JMqBuC,2BAA2B,CAA3B,I/JNqBvC,C;U+JOqBY,mBAAY,CAAZ,C;U/JPqBZ,I+JOqBkB,kB/JPqBlB,E+JOqBuC,2BAA2B,CAA3B,I/JPqBvC,C;U+JQqBY,mBAAY,CAAZ,C;U/JRqBZ,I+JQqBkB,kB/JRqBlB,E+JQqBuC,2BAA2B,CAA3B,I/JRqBvC,C;U+JSqBY,mBAAY,CAAZ,C;U/JTqBZ,I+JSqBkB,kB/JTqBlB,E+JSqBuC,2BAA2B,CAA3B,I/JTqBvC,C;U+JUqBY,mBAAY,CAAZ,C;U/JVqBZ,I+JUqBkB,kB/JVqBlB,E+JUqBuC,2BAA2B,CAA3B,I/JVqBvC,C;U+JWqBY,mBAAY,CAAZ,C;U/JXqBZ,I+JWqBkB,kB/JXqBlB,E+JWqBuC,2BAA2B,CAA3B,I/JXqBvC,C;U+JYqBY,mBAAY,CAAZ,C;U/JZqBZ,I+JYqBkB,kB/JZqBlB,E+JYqBuC,2BAA2B,CAA3B,I/JZqBvC,C;U+JaqBY,mBAAY,CAAZ,C;U/JbqBZ,I+JaqBkB,kB/JbqBlB,E+JaqBuC,2BAA2B,CAA3B,I/JbqBvC,C;U+JcqBY,mBAAY,CAAZ,C;U/JdqBZ,I+JcqBkB,kB/JdqBlB,E+JcqBuC,2BAA2B,CAA3B,I/JdqBvC,C;U+JeqBY,mBAAY,CAAZ,C;U/JfqBZ,I+JeqBkB,kB/JfqBlB,E+JeqBuC,2BAA2B,CAA3B,I/JfqBvC,C;U+JgqBY,mBAAY,CAAZ,C;U/JhqBZ,I+JgqBkB,kB/JhqBlB,E+JgqBuC,2BAA2B,CAA3B,I/JhqBvC,C;U+JiqBY,mBAAY,CAAZ,C;U/JjqBZ,I+JiqBkB,kB/JjqBlB,E+JiqBuC,2BAA2B,CAA3B,I/JjqBvC,C;U+JkqBY,mBAAY,CAAZ,C;U/JlqBZ,I+JkqBkB,kB/JlqBlB,E+JkqBuC,2BAA2B,CAA3B,I/JlqBvC,C;U+JmqBY,mBAAY,CAAZ,C;U/JnqBZ,I+JmqBkB,kB/JnqBlB,E+JmqBuC,2BAA2B,CAA3B,I/JnqBvC,C;U+JoqBY,mBAAY,CAAZ,C;U/JpqBZ,I+JoqBkB,kB/JpqBlB,E+JoqBuC,2BAA2B,CAA3B,I/JpqBvC,C;U+JqqBY,mBAAY,CAAZ,C;U/JrqBZ,I+JqqBkB,kB/JrqBlB,E+JqqBuC,2BAA2B,CAA3B,I/JrqBvC,C;U+JsqBY,mBAAY,CAAZ,C;U/JtqBZ,I+JsqBkB,kB/JtqBlB,E+JsqBuC,2BAA2B,CAA3B,I/JtqBvC,C;U+JuqBY,mBAAY,CAAZ,C;U/JvqBZ,I+JuqBkB,kB/JvqBlB,E+JuqBuC,2BAA2B,CAA3B,I/JvqBvC,C;U+JwqBY,mBAAY,CAAZ,C;U/JxqBZ,I+JwqBkB,kB/JxqBlB,E+JwqBuC,2BAA2B,CAA3B,I/JxqBvC,C;U+JyqBY,mBAAY,CAAZ,C;U/JzqBZ,I+JyqBkB,kB/JzqBlB,E+JyqBuC,2BAA2B,CAA3B,I/JzqBvC,C;U+J0qBY,mBAAY,CAAZ,C;U/J1qBZ,I+J0qBkB,kB/J1qBlB,E+J0qBuC,2BAA2B,CAA3B,I/J1qBvC,C;U+J2qBY,mBAAY,CAAZ,C;U/J3qBZ,I+J2qBkB,kB/J3qBlB,E+J2qBuC,2BAA2B,CAA3B,I/J3qBvC,C;U+J4qBY,mBAAY,CAAZ,C;U/J5qBZ,I+J4qBkB,kB/J5qBlB,E+J4qBuC,2BAA2B,CAA3B,I/J5qBvC,C;U+J6qBY,mBAAY,CAAZ,C;U/J7qBZ,I+J6qBkB,kB/J7qBlB,E+J6qBuC,2BAA2B,CAA3B,I/J7qBvC,C;U+J8qBY,mBAAY,CAAZ,C;U/J9qBZ,I+J8qBkB,kB/J9qBlB,E+J8qBuC,2BAA2B,CAA3B,I/J9qBvC,C;U+J+qBY,mBAAY,CAAZ,C;U/J/qBZ,I+J+qBkB,kB/J/qBlB,E+J+qBuC,2BAA2B,CAA3B,I/J/qBvC,C;U+JgrBY,mBAAY,CAAZ,C;U/JhrBZ,I+JgrBkB,kB/JhrBlB,E+JgrBuC,2BAA2B,CAA3B,I/JhrBvC,C;U+JirBY,mBAAY,CAAZ,C;U/JjrBZ,I+JirBkB,kB/JjrBlB,E+JirBuC,2BAA2B,CAA3B,I/JjrBvC,C;U+JkrBY,mBAAY,CAAZ,C;U/JlrBZ,I+JkrBkB,kB/JlrBlB,E+JkrBuC,2BAA2B,CAA3B,I/JlrBvC,C;U+JmrBY,mBAAY,CAAZ,C;U/JnrBZ,I+JmrBkB,kB/JnrBlB,E+JmrBuC,2BAA2B,CAA3B,I/JnrBvC,C;U+JorBY,mBAAY,CAAZ,C;U/JprBZ,I+JorBkB,kB/JprBlB,E+JorBuC,2BAA2B,CAA3B,I/JprBvC,C;U+JqrBY,mBAAY,CAAZ,C;U/JrrBZ,I+JqrBkB,kB/JrrBlB,E+JqrBuC,2BAA2B,CAA3B,I/JrrBvC,C;U+JsrBY,mBAAY,CAAZ,C;U/JtrBZ,I+JsrBkB,kB/JtrBlB,E+JsrBuC,2BAA2B,CAA3B,I/JtrBvC,C;U+JurBY,mBAAY,CAAZ,C;U/JvrBZ,I+JurBkB,kB/JvrBlB,E+JurBuC,2BAA2B,CAA3B,I/JvrBvC,C;U+JwrBY,mBAAY,CAAZ,C;U/JxrBZ,I+JwrBkB,kB/JxrBlB,E+JwrBuC,2BAA2B,CAA3B,I/JxrBvC,C;U+JyrBY,mBAAY,CAAZ,C;U/JzrBZ,I+JyrBkB,kB/JzrBlB,E+JyrBuC,2BAA2B,CAA3B,I/JzrBvC,C;U+J0rBY,mBAAY,CAAZ,C;U/J1rBZ,I+J0rBkB,kB/J1rBlB,E+J0rBuC,2BAA2B,CAA3B,I/J1rBvC,C;U+J2rBY,mBAAY,CAAZ,C;U/J3rBZ,I+J2rBkB,kB/J3rBlB,E+J2rBuC,2BAA2B,CAA3B,I/J3rBvC,C;U+J4rBY,mBAAY,CAAZ,C;U/J5rBZ,I+J4rBkB,kB/J5rBlB,E+J4rBuC,2BAA2B,CAA3B,I/J5rBvC,C;U+J6rBY,mBAAY,CAAZ,C;U/J7rBZ,I+J6rBkB,kB/J7rBlB,E+J6rBuC,2BAA2B,CAA3B,I/J7rBvC,C;U+J8rBY,mBAAY,CAAZ,C;U/J9rBZ,I+J8rBkB,kB/J9rBlB,E+J8rBuC,2BAA2B,CAA3B,I/J9rBvC,C;U+J+rBY,mBAAY,CAAZ,C;U/J/rBZ,I+J+rBkB,kB/J/rBlB,E+J+rBuC,2BAA2B,CAA3B,I/J/rBvC,C;U+JgsBY,mBAAY,CAAZ,C;U/JhsBZ,I+JgsBkB,kB/JhsBlB,E+JgsBuC,2BAA2B,CAA3B,I/JhsBvC,C;U+JisBY,mBAAY,CAAZ,C;U/JjsBZ,I+JisBkB,kB/JjsBlB,E+JisBuC,2BAA2B,CAA3B,I/JjsBvC,C;U+JksBY,mBAAY,CAAZ,C;U/JlsBZ,I+JksBkB,kB/JlsBlB,E+JksBuC,2BAA2B,CAA3B,I/JlsBvC,C;U+JmsBY,mBAAY,CAAZ,C;U/JnsBZ,I+JmsBkB,kB/JnsBlB,E+JmsBuC,2BAA2B,CAA3B,I/JnsBvC,C;U+JosBY,mBAAY,CAAZ,C;U/JpsBZ,I+JosBkB,kB/JpsBlB,E+JosBuC,2BAA2B,CAA3B,I/JpsBvC,C;U+JqsBY,mBAAY,CAAZ,C;U/JrsBZ,I+JqsBkB,kB/JrsBlB,E+JqsBuC,2BAA2B,CAA3B,I/JrsBvC,C;U+JssBY,mBAAY,CAAZ,C;U/JtsBZ,I+JssBkB,kB/JtsBlB,E+JssBuC,2BAA2B,CAA3B,I/JtsBvC,C;U+JusBY,mBAAY,CAAZ,C;U/JvsBZ,I+JusBkB,kB/JvsBlB,E+JusBuC,2BAA2B,CAA3B,I/JvsBvC,C;U+JwsBY,mBAAY,CAAZ,C;U/JxsBZ,I+JwsBkB,kB/JxsBlB,E+JwsBuC,2BAA2B,CAA3B,I/JxsBvC,C;U+JysBY,mBAAY,CAAZ,C;U/JzsBZ,I+JysBkB,kB/JzsBlB,E+JysBuC,2BAA2B,CAA3B,I/JzsBvC,C;U+J0sBY,mBAAY,CAAZ,C;U/J1sBZ,I+J0sBkB,kB/J1sBlB,E+J0sBuC,2BAA2B,CAA3B,I/J1sBvC,C;U+J2sBY,mBAAY,CAAZ,C;U/J3sBZ,I+J2sBkB,kB/J3sBlB,E+J2sBuC,2BAA2B,CAA3B,I/J3sBvC,C;U+J4sBY,mBAAY,CAAZ,C;U/J5sBZ,I+J4sBkB,kB/J5sBlB,E+J4sBuC,2BAA2B,CAA3B,I/J5sBvC,C;U+J6sBY,mBAAY,CAAZ,C;U/J7sBZ,I+J6sBkB,kB/J7sBlB,E+J6sBuC,2BAA2B,CAA3B,I/J7sBvC,C;U+J8sBY,mBAAY,CAAZ,C;U/J9sBZ,I+J8sBkB,kB/J9sBlB,E+J8sBuC,2BAA2B,CAA3B,I/J9sBvC,C;U+J+sBY,mBAAY,CAAZ,C;U/J/sBZ,I+J+sBkB,kB/J/sBlB,E+J+sBuC,2BAA2B,CAA3B,I/J/sBvC,C;U+JgtBY,mBAAY,CAAZ,C;U/JhtBZ,I+JgtBkB,kB/JhtBlB,E+JgtBuC,2BAA2B,CAA3B,I/JhtBvC,C;U+JitBY,mBAAY,CAAZ,C;U/JjtBZ,I+JitBkB,kB/JjtBlB,E+JitBuC,2BAA2B,CAA3B,I/JjtBvC,C;U+JktBY,mBAAY,CAAZ,C;U/JltBZ,I+JktBkB,kB/JltBlB,E+JktBuC,2BAA2B,CAA3B,I/JltBvC,C;U+JmtBY,mBAAY,CAAZ,C;U/JntBZ,I+JmtBkB,kB/JntBlB,E+JmtBuC,2BAA2B,CAA3B,I/JntBvC,C;U+JotBY,mBAAY,CAAZ,C;U/JptBZ,I+JotBkB,kB/JptBlB,E+JotBuC,2BAA2B,CAA3B,I/JptBvC,C;U+JqtBY,mBAAY,CAAZ,C;U/JrtBZ,I+JqtBkB,kB/JrtBlB,E+JqtBuC,2BAA2B,CAA3B,I/JrtBvC,C;U+JstBY,mBAAY,CAAZ,C;U/JttBZ,I+JstBkB,kB/JttBlB,E+JstBuC,2BAA2B,CAA3B,I/JttBvC,C;U+JutBY,mBAAY,CAAZ,C;U/JvtBZ,I+JutBkB,kB/JvtBlB,E+JutBuC,2BAA2B,CAA3B,I/JvtBvC,C;U+JwtBY,mBAAY,CAAZ,C;U/JxtBZ,I+JwtBkB,kB/JxtBlB,E+JwtBuC,2BAA2B,CAA3B,I/JxtBvC,C;U+JytBY,mBAAY,CAAZ,C;U/JztBZ,I+JytBkB,kB/JztBlB,E+JytBuC,2BAA2B,CAA3B,I/JztBvC,C;U+J0tBY,mBAAY,CAAZ,C;U/J1tBZ,I+J0tBkB,kB/J1tBlB,E+J0tBuC,2BAA2B,CAA3B,I/J1tBvC,C;U+J2tBY,mBAAY,CAAZ,C;U/J3tBZ,I+J2tBkB,kB/J3tBlB,E+J2tBuC,2BAA2B,CAA3B,I/J3tBvC,C;U+J4tBY,mBAAY,CAAZ,C;U/J5tBZ,I+J4tBkB,kB/J5tBlB,E+J4tBuC,2BAA2B,CAA3B,I/J5tBvC,C;U+J6tBY,mBAAY,CAAZ,C;U/J7tBZ,I+J6tBkB,kB/J7tBlB,E+J6tBuC,2BAA2B,CAA3B,I/J7tBvC,C;U+J8tBY,mBAAY,CAAZ,C;U/J9tBZ,I+J8tBkB,kB/J9tBlB,E+J8tBuC,2BAA2B,CAA3B,I/J9tBvC,C;U+J+tBY,mBAAY,CAAZ,C;U/J/tBZ,I+J+tBkB,kB/J/tBlB,E+J+tBuC,2BAA2B,CAA3B,I/J/tBvC,C;U+JguBY,mBAAY,CAAZ,C;U/JhuBZ,I+JguBkB,kB/JhuBlB,E+JguBuC,2BAA2B,CAA3B,I/JhuBvC,C;U+JiuBY,mBAAY,CAAZ,C;U/JjuBZ,I+JiuBkB,kB/JjuBlB,E+JiuBuC,2BAA2B,CAA3B,I/JjuBvC,C;U+JkuBY,mBAAY,CAAZ,C;U/JluBZ,I+JkuBkB,kB/JluBlB,E+JkuBuC,2BAA2B,CAA3B,I/JluBvC,C;U+JmuBY,mBAAY,CAAZ,C;U/JnuBZ,I+JmuBkB,kB/JnuBlB,E+JmuBuC,2BAA2B,CAA3B,I/JnuBvC,C;U+JouBY,mBAAY,CAAZ,C;U/JpuBZ,I+JouBkB,kB/JpuBlB,E+JouBuC,2BAA2B,CAA3B,I/JpuBvC,C;U+JquBY,mBAAY,CAAZ,C;U/JruBZ,I+JquBkB,kB/JruBlB,E+JquBuC,2BAA2B,CAA3B,I/JruBvC,C;U+JsuBY,mBAAY,CAAZ,C;U/JtuBZ,I+JsuBkB,kB/JtuBlB,E+JsuBuC,2BAA2B,CAA3B,I/JtuBvC,C;U+JuuBY,mBAAY,CAAZ,C;U/JvuBZ,I+JuuBkB,kB/JvuBlB,E+JuuBuC,2BAA2B,CAA3B,I/JvuBvC,C;U+JwuBY,mBAAY,CAAZ,C;U/JxuBZ,I+JwuBkB,kB/JxuBlB,E+JwuBuC,2BAA2B,CAA3B,I/JxuBvC,C;U+JyuBY,mBAAY,CAAZ,C;U/JzuBZ,I+JyuBkB,kB/JzuBlB,E+JyuBuC,2BAA2B,CAA3B,I/JzuBvC,C;U+J0uBY,mBAAY,CAAZ,C;U/J1uBZ,I+J0uBkB,kB/J1uBlB,E+J0uBuC,2BAA2B,CAA3B,I/J1uBvC,C;U+J2uBY,mBAAY,CAAZ,C;U/J3uBZ,I+J2uBkB,kB/J3uBlB,E+J2uBuC,2BAA2B,CAA3B,I/J3uBvC,C;U+J4uBY,mBAAY,CAAZ,C;U/J5uBZ,I+J4uBkB,kB/J5uBlB,E+J4uBuC,2BAA2B,CAA3B,I/J5uBvC,C;U+J6uBY,mBAAY,CAAZ,C;U/J7uBZ,I+J6uBkB,kB/J7uBlB,E+J6uBuC,2BAA2B,CAA3B,I/J7uBvC,C;U+J8uBY,mBAAY,CAAZ,C;U/J9uBZ,I+J8uBkB,kB/J9uBlB,E+J8uBuC,2BAA2B,CAA3B,I/J9uBvC,C;U+J+uBY,mBAAY,CAAZ,C;U/J/uBZ,I+J+uBkB,kB/J/uBlB,E+J+uBuC,2BAA2B,CAA3B,I/J/uBvC,C;U+JgvBY,mBAAY,CAAZ,C;U/JhvBZ,I+JgvBkB,kB/JhvBlB,E+JgvBuC,2BAA2B,CAA3B,I/JhvBvC,C;U+JivBY,mBAAY,CAAZ,C;U/JjvBZ,I+JivBkB,kB/JjvBlB,E+JivBuC,2BAA2B,CAA3B,I/JjvBvC,C;U+JkvBY,mBAAY,CAAZ,C;U/JlvBZ,I+JkvBkB,kB/JlvBlB,E+JkvBuC,2BAA2B,CAA3B,I/JlvBvC,C;U+JmvBY,mBAAY,CAAZ,C;U/JnvBZ,I+JmvBkB,kB/JnvBlB,E+JmvBuC,2BAA2B,CAA3B,I/JnvBvC,C;U+JovBY,mBAAY,CAAZ,C;U/JpvBZ,I+JovBkB,kB/JpvBlB,E+JovBuC,2BAA2B,CAA3B,I/JpvBvC,C;U+JqvBY,mBAAY,CAAZ,C;U/JrvBZ,I+JqvBkB,kB/JrvBlB,E+JqvBuC,2BAA2B,CAA3B,I/JrvBvC,C;U+JsvBY,mBAAY,CAAZ,C;U/JtvBZ,I+JsvBkB,kB/JtvBlB,E+JsvBuC,2BAA2B,CAA3B,I/JtvBvC,C;U+JuvBY,mBAAY,CAAZ,C;U/JvvBZ,I+JuvBkB,kB/JvvBlB,E+JuvBuC,2BAA2B,CAA3B,I/JvvBvC,C;U+JwvBY,mBAAY,CAAZ,C;U/JxvBZ,I+JwvBkB,kB/JxvBlB,E+JwvBuC,2BAA2B,CAA3B,I/JxvBvC,C;U+JyvBY,mBAAY,CAAZ,C;U/JzvBZ,I+JyvBkB,kB/JzvBlB,E+JyvBuC,2BAA2B,CAA3B,I/JzvBvC,C;U+J0vBY,mBAAY,CAAZ,C;U/J1vBZ,I+J0vBkB,kB/J1vBlB,E+J0vBuC,2BAA2B,CAA3B,I/J1vBvC,C;U+J2vBY,mBAAY,CAAZ,C;U/J3vBZ,I+J2vBkB,kB/J3vBlB,E+J2vBuC,2BAA2B,CAA3B,I/J3vBvC,C;U+J4vBY,mBAAY,CAAZ,C;U/J5vBZ,I+J4vBkB,kB/J5vBlB,E+J4vBuC,2BAA2B,CAA3B,I/J5vBvC,C;U+J6vBY,mBAAY,CAAZ,C;U/J7vBZ,I+J6vBkB,kB/J7vBlB,E+J6vBuC,2BAA2B,CAA3B,I/J7vBvC,C;U+J8vBY,mBAAY,CAAZ,C;U/J9vBZ,I+J8vBkB,kB/J9vBlB,E+J8vBuC,2BAA2B,CAA3B,I/J9vBvC,C;U+J+vBY,mBAAY,CAAZ,C;U/J/vBZ,I+J+vBkB,kB/J/vBlB,E+J+vBuC,2BAA2B,CAA3B,I/J/vBvC,C;U+JgwBY,mBAAY,CAAZ,C;U/JhwBZ,I+JgwBkB,kB/JhwBlB,E+JgwBuC,2BAA2B,CAA3B,I/JhwBvC,C;U+JiwBY,mBAAY,CAAZ,C;U/JjwBZ,I+JiwBkB,kB/JjwBlB,E+JiwBuC,2BAA2B,CAA3B,I/JjwBvC,C;U+JkwBY,mBAAY,CAAZ,C;U/JlwBZ,I+JkwBkB,kB/JlwBlB,E+JkwBuC,2BAA2B,CAA3B,I/JlwBvC,C;U+JmwBY,mBAAY,CAAZ,C;U/JnwBZ,I+JmwBkB,kB/JnwBlB,E+JmwBuC,2BAA2B,CAA3B,I/JnwBvC,C;U+JowBY,mBAAY,CAAZ,C;U/JpwBZ,I+JowBkB,kB/JpwBlB,E+JowBuC,2BAA2B,CAA3B,I/JpwBvC,C;U+JqwBY,mBAAY,CAAZ,C;U/JrwBZ,I+JqwBkB,kB/JrwBlB,E+JqwBuC,2BAA2B,CAA3B,I/JrwBvC,C;U+JswBY,mBAAY,CAAZ,C;U/JtwBZ,I+JswBkB,kB/JtwBlB,E+JswBuC,2BAA2B,CAA3B,I/JtwBvC,C;U+JuwBY,mBAAY,CAAZ,C;U/JvwBZ,I+JuwBkB,kB/JvwBlB,E+JuwBuC,2BAA2B,CAA3B,I/JvwBvC,C;U+JwwBY,mBAAY,CAAZ,C;U/JxwBZ,I+JwwBkB,kB/JxwBlB,E+JwwBuC,2BAA2B,CAA3B,I/JxwBvC,C;U+JywBY,mBAAY,CAAZ,C;U/JzwBZ,I+JywBkB,kB/JzwBlB,E+JywBuC,2BAA2B,CAA3B,I/JzwBvC,C;U+J0wBY,mBAAY,CAAZ,C;U/J1wBZ,I+J0wBkB,kB/J1wBlB,E+J0wBuC,2BAA2B,CAA3B,I/J1wBvC,C;U+J2wBY,mBAAY,CAAZ,C;U/J3wBZ,I+J2wBkB,kB/J3wBlB,E+J2wBuC,2BAA2B,CAA3B,I/J3wBvC,C;U+J4wBY,mBAAY,CAAZ,C;U/J5wBZ,I+J4wBkB,kB/J5wBlB,E+J4wBuC,2BAA2B,CAA3B,I/J5wBvC,C;U+J6wBY,mBAAY,CAAZ,C;U/J7wBZ,I+J6wBkB,kB/J7wBlB,E+J6wBuC,2BAA2B,CAA3B,I/J7wBvC,C;U+J8wBY,mBAAY,CAAZ,C;U/J9wBZ,I+J8wBkB,kB/J9wBlB,E+J8wBuC,2BAA2B,CAA3B,I/J9wBvC,C;U+J+wBY,mBAAY,CAAZ,C;U/J/wBZ,I+J+wBkB,kB/J/wBlB,E+J+wBuC,2BAA2B,CAA3B,I/J/wBvC,C;U+JgxBY,mBAAY,CAAZ,C;U/JhxBZ,I+JgxBkB,kB/JhxBlB,E+JgxBuC,2BAA2B,CAA3B,I/JhxBvC,C;U+JixBY,mBAAY,CAAZ,C;U/JjxBZ,I+JixBkB,kB/JjxBlB,E+JixBuC,2BAA2B,CAA3B,I/JjxBvC,C;U+JkxBY,mBAAY,CAAZ,C;U/JlxBZ,I+JkxBkB,kB/JlxBlB,E+JkxBuC,2BAA2B,CAA3B,I/JlxBvC,C;U+JmxBY,mBAAY,CAAZ,C;U/JnxBZ,I+JmxBkB,kB/JnxBlB,E+JmxBuC,2BAA2B,CAA3B,I/JnxBvC,C;U+JoxBY,mBAAY,CAAZ,C;U/JpxBZ,I+JoxBkB,kB/JpxBlB,E+JoxBuC,2BAA2B,CAA3B,I/JpxBvC,C;U+JqxBY,mBAAY,CAAZ,C;U/JrxBZ,I+JqxBkB,kB/JrxBlB,E+JqxBuC,2BAA2B,CAA3B,I/JrxBvC,C;U+JsxBY,mBAAY,CAAZ,C;U/JtxBZ,I+JsxBkB,kB/JtxBlB,E+JsxBuC,2BAA2B,CAA3B,I/JtxBvC,C;U+JuxBY,mBAAY,CAAZ,C;U/JvxBZ,I+JuxBkB,kB/JvxBlB,E+JuxBuC,2BAA2B,CAA3B,I/JvxBvC,C;U+JwxBY,mBAAY,CAAZ,C;U/JxxBZ,I+JwxBkB,kB/JxxBlB,E+JwxBuC,2BAA2B,CAA3B,I/JxxBvC,C;U+JyxBY,mBAAY,CAAZ,C;U/JzxBZ,I+JyxBkB,kB/JzxBlB,E+JyxBuC,2BAA2B,CAA3B,I/JzxBvC,C;U+J0xBY,mBAAY,CAAZ,C;U/J1xBZ,I+J0xBkB,kB/J1xBlB,E+J0xBuC,2BAA2B,CAA3B,I/J1xBvC,C;U+J2xBY,mBAAY,CAAZ,C;U/J3xBZ,I+J2xBkB,kB/J3xBlB,E+J2xBuC,2BAA2B,CAA3B,I/J3xBvC,C;U+J4xBY,mBAAY,CAAZ,C;U/J5xBZ,I+J4xBkB,kB/J5xBlB,E+J4xBuC,2BAA2B,CAA3B,I/J5xBvC,C;U+J6xBY,mBAAY,CAAZ,C;U/J7xBZ,I+J6xBkB,kB/J7xBlB,E+J6xBuC,2BAA2B,CAA3B,I/J7xBvC,C;U+J8xBY,mBAAY,CAAZ,C;U/J9xBZ,I+J8xBkB,kB/J9xBlB,E+J8xBuC,2BAA2B,CAA3B,I/J9xBvC,C;U+J+xBY,mBAAY,CAAZ,C;U/J/xBZ,I+J+xBkB,kB/J/xBlB,E+J+xBuC,2BAA2B,CAA3B,I/J/xBvC,C;U+JgyBY,mBAAY,CAAZ,C;U/JhyBZ,I+JgyBkB,kB/JhyBlB,E+JgyBuC,2BAA2B,CAA3B,I/JhyBvC,C;U+JiyBY,mBAAY,CAAZ,C;U/JjyBZ,I+JiyBkB,kB/JjyBlB,E+JiyBuC,2BAA2B,CAA3B,I/JjyBvC,C;U+JkyBY,mBAAY,CAAZ,C;U/JlyBZ,I+JkyBkB,kB/JlyBlB,E+JkyBuC,2BAA2B,CAA3B,I/JlyBvC,C;U+JmyBY,mBAAY,CAAZ,C;U/JnyBZ,I+JmyBkB,kB/JnyBlB,E+JmyBuC,2BAA2B,CAA3B,I/JnyBvC,C;U+JoyBY,mBAAY,CAAZ,C;U/JpyBZ,I+JoyBkB,kB/JpyBlB,E+JoyBuC,2BAA2B,CAA3B,I/JpyBvC,C;U+JqyBY,mBAAY,CAAZ,C;U/JryBZ,I+JqyBkB,kB/JryBlB,E+JqyBuC,2BAA2B,CAA3B,I/JryBvC,C;U+JsyBY,mBAAY,CAAZ,C;U/JtyBZ,I+JsyBkB,kB/JtyBlB,E+JsyBuC,2BAA2B,CAA3B,I/JtyBvC,C;U+JuyBY,mBAAY,CAAZ,C;U/JvyBZ,I+JuyBkB,kB/JvyBlB,E+JuyBuC,2BAA2B,CAA3B,I/JvyBvC,C;U+JwyBY,mBAAY,CAAZ,C;U/JxyBZ,I+JwyBkB,kB/JxyBlB,E+JwyBuC,2BAA2B,CAA3B,I/JxyBvC,C;U+JyyBY,mBAAY,CAAZ,C;U/JzyBZ,I+JyyBkB,kB/JzyBlB,E+JyyBuC,2BAA2B,CAA3B,I/JzyBvC,C;U+J0yBY,mBAAY,CAAZ,C;U/J1yBZ,I+J0yBkB,kB/J1yBlB,E+J0yBuC,2BAA2B,CAA3B,I/J1yBvC,C;U+J2yBY,mBAAY,CAAZ,C;U/J3yBZ,I+J2yBkB,kB/J3yBlB,E+J2yBuC,2BAA2B,CAA3B,I/J3yBvC,C;U+J4yBY,mBAAY,CAAZ,C;U/J5yBZ,I+J4yBkB,kB/J5yBlB,E+J4yBuC,2BAA2B,CAA3B,I/J5yBvC,C;U+J6yBY,mBAAY,CAAZ,C;U/J7yBZ,I+J6yBkB,kB/J7yBlB,E+J6yBuC,2BAA2B,CAA3B,I/J7yBvC,C;U+J8yBY,mBAAY,CAAZ,C;U/J9yBZ,I+J8yBkB,kB/J9yBlB,E+J8yBuC,2BAA2B,CAA3B,I/J9yBvC,C;U+J+yBY,mBAAY,CAAZ,C;U/J/yB

J1jBuF,kBAAY,OAAZ,GAAmB,YAAnB,I/J0jBvF,C;;;Q+JvjBQ,8BAAuB,aAAvB,EAAsC,QAAtC,C;;MAGJ,OA
AO,I;K;uCAGX,iB;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,SAAzB,C;MAjRN,Q;MAmRP,OAnRO,2BAQyC,m
BAAY,cA2Q3B,KA3Q2B,IAAZ,CARzC,4D;K;uCAsRX,0B;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,SAAzB,C;
MAEb,oBAjRgD,mBAAY,cAiR1B,KAjR0B,IAAZ,C;MARzC,Q;MA0RP,iBA1RO,2BA0RsB,aA1RtB,4D;MA2R
P,mBAAY,aAAZ,IAA6B,O;MAE7B,OAAO,U;K;0CAGX,mB;MAAoD,0BAAQ,OAAR,MAAoB,E;K;yCAExE,m
B;MAIsB,IAIA,IAJA,EAIuB,M;MAPzC,WA3RgD,mBAAY,cA2RnC,SA3RmC,IAAZ,C;MA6RhD,IAAI,cAAO,I
AAX,C;QACI,iBAAc,WAAd,UAAyB,IAAzB,U;UACI,IAAI,gBAAW,mBAAY,KAAZ,CAAX,CAAJ,C;YAAmC,
OAAO,QAAQ,WAAR,I;;aAE3C,IAAI,eAAQ,IAAZ,C;QACW,kB;QAAuB,SAAZ,kBAAY,O;QAArC,qD;UACI,I
AAI,gBAAW,mBAAY,OAAZ,CAAX,CAAJ,C;YAAmC,OAAO,UAAQ,WAAR,I;;QAE9C,mBAAc,CAAd,YAAs
B,IAAtB,Y;UACI,IAAI,gBAAW,mBAAY,OAAZ,CAAX,CAAJ,C;YAAmC,OAAO,UAAQ,kBAAY,OAApB,GA
A2B,WAA3B,I;;;MAIlD,OAAO,E;K;6CAGX,mB;MAIsC,UAOJ,MAPI,EAOa,M;MAV/C,WA9SgD,cA8
SnC,SA9SmC,IAAZ,C;MAgThD,IAAI,cAAO,IAAX,C;QACkC,kB;QAA9B,iBAAc,OAAO,CAAP,IAAd,yB;UAC
I,IAAI,gBAAW,mBAAY,KAAZ,CAAX,CAAJ,C;YAAmC,OAAO,QAAQ,WAAR,I;;aAE3C,IAAI,cAAO,IAAX,C
;QACH,mBAAc,OAAO,CAAP,IAAd,aAA8B,CAA9B,Y;UACI,IAAI,gBAAW,mBAAY,OAAZ,CAAX,CAAJ,C;Y
AAmC,OAAO,UAAQ,kBAAY,OAApB,GAA2B,WAA3B,I;;QAEpB,uBAAZ,kBAAY,C;QAAiB,oB;QAA3C,wD;
UACI,IAAI,gBAAW,mBAAY,OAAZ,CAAX,CAAJ,C;YAAmC,OAAO,UAAQ,WAAR,I;;;MAIlD,OAAO,E;K;wC
AGX,mB;MACI,YAAY,mBAAQ,OAAR,C;MACZ,IAAI,UAAS,EAAb,C;QAAiB,OAAO,K;MACxB,sBAAS,KA
AT,C;MACA,OAAO,I;K;4CAGX,iB;MACI,oCAAa,2BAAkB,KAAlB,EAAyB,SAAzB,C;MAEb,IAAI,UAAS,sBA
Ab,C;QACI,OAAO,iB;aACJ,IAAI,UAAS,CAAb,C;QACH,OAAO,kB;;MAGX,oBAhVgD,mBAAY,cAgV1B,KAh
V0B,IAAZ,C;MARzC,Q;MAyVP,cAzVO,2BAyVsB,aAzVtB,4D;MA2VP,IAAI,QAAQ,aAAS,CAArB,C;QAEI,I
AAI,iBAAiB,WAArB,C;U/JoeR,U+JneY,kB/JmeZ,E+JneiC,kB/JmejC,E+Jne8C,cAAO,CAAP,IAA9C,I/Jme9C,E+JnewD,
W/JmexD,E+Jne8D,a/Jme9D,C;;UAAA,U+JjeY,kB/JieZ,E+JjeiC,kB/JiejC,E+Jje8C,C/Jie9C,E+JjeiD,C/JiejD,E+Jje
oD,a/JiepD,C;U+JheY,mBAAY,CAAZ,IAAiB,mBAAY,kBAAY,OAAZ,GAAmB,CAAnB,IAAZ,C;U/Jge7B,U+J/d
Y,kB/J+dZ,E+J/diC,kB/J+djC,E+J/d8C,cAAO,CAAP,IAA9C,I/J+d9C,E+J/dwD,W/J+dxD,E+J/d8D,kBAAY,OAAZ,GAA
mB,CAAnB,I/J+d9D,C;;Q+J5dQ,mBAAY,WAAZ,IAAoB,I;QACpB,cAAO,mBAAY,WAAZ,C;;QAGP,wBAjW4C
,mBAAY,cAiWlB,sBAjWkB,IAAZ,C;QAmW5C,IAAI,iBAAiB,iBAArB,C;U/JsdR,U+JrdY,kB/JqdZ,E+JrdiC,kB/J
qdjC,E+Jrd8C,a/Jqd9C,E+Jrd6D,gBAAgB,CAAhB,I/Jqd7D,E+JrdgF,oBAAoB,CAApB,I/JqdhF,C;;UAAA,U+JndY
,kB/JmdZ,E+JndiC,kB/JmdjC,E+Jnd8C,a/Jmd9C,E+Jnd6D,gBAAgB,CAAhB,I/Jmd7D,E+JndgF,kBAAY,O/Jmd5F
,C;U+JldY,mBAAY,kBAAY,OAAZ,GAAmB,CAAnB,IAAZ,C;U/JkdhD,U+JjdY,kB/JidZ
,E+JjdiC,kB/JidjC,E+Jjd8C,C/Jid9C,E+JjdiD,C/JidjD,E+JjdoD,oBAAoB,CAApB,I/JidpD,C;;Q+J9cQ,mBAAY,iBA
AZ,IAAiC,I;;MAErC,wBAAQ,CAAR,I;MAEA,OAAO,O;K;6CAGX,oB;MAAkE,0B;;QAa5C,wD;QART,aAAL,I
AAK,U;QAAL,Y;UAA8B,SAAZ,kB9K6wOnB,YAAQ,C;;Q8K7wOX,W;UACI,yBAAO,K;UAAP,2B;;QAEJ,WA
1XgD,mBAAY,cA0XnC,SA1XmC,IAAZ,C;QA2XhD,cAAc,W;QACd,eAAe,K;QAEf,IAAI,cAAO,IAAX,C;UACI,
iBAAc,WAAd,UAAyB,IAAzB,U;YACI,cAAc,mBAAY,KAAZ,C;YAAmC,gBAAW,CAAX,IAAnC,wBAAiE,0EAAjBkE,CAi
BhF,C;cACI,mBAAY,gBAAZ,EAAyB,wBAAZ,YAAyB,O;;cAEtC,WAAW,I;;UAGP,OAAZ,kBAAY,EAAZ,IAAL,I
AAL,EAAW,OAAX,EAAoB,IAApB,B,C;;UAEJ,oB;UAAuB

UAEV,mBAAc,CAAd,YAAsB,IAAtB,Y;YACI,gBAAc,mBAAY,OAAZ,C;YACd,mBAAY,OAAZ,IAAqB,I;YAGr
B,IA1C+E,wBA0CjE,kFA1CiE,CA0C/E,C;cACI,mBAAY,OAAZ,IAAuB,S;cACvB,UAAU,mBAAY,OAAZ,C;;cA
EV,WAAW,I;;;;QAIvB,IAAI,QAAJ,C;UACI,YAAO,mBAAY,UAAU,WAAV,IAAZ,C;QAEX,yBAAO,Q;;;MArD
uD,6B;K;2CAElE,qB;MASsB,IAII,IAJJ,EAKM,MALN,EAaA,MAbA,EAauB,MAbvB,EAkBI,MAlBJ,EAmBM,M
AnBN,EA+BI,M;MAvCb,aAAL,IAAK,U;MAAL,Y;QAA8B,SAAZ,kB9K6wOnB,YAAQ,C;;M8K7wOX,W;QACI
,OAAO,K;MAEX,WA1XgD,mBAAY,cA0XnC,SA1XmC,IAAZ,C;MA2XhD,cAAc,W;MACd,eAAe,K;MAEf,IAA
I,cAAO,IAAX,C;QACI,iBAAc,WAAd,UAAyB,IAAzB,U;UACI,cAAc,mBAAY,KAAZ,C;UAGd,IAAI,UAAU,0E
AAV,CAAJ,C;YACI,mBAAY,gBAAZ,EAAY,wBAAZ,YAAyB,O;;YAEzB,WAAW,I;;QAGP,OAAZ,kBAAY,EA
AK,IAAL,EAAW,OAAX,EAAoB,IAApB,C;;QAGE,oB;QAAuB,SAAZ,kBAAY,O;QAArC,uD;UACI,gBAAc,mB
AAY,OAAZ,C;UACd,mBAAY,OAAZ,IAAqB,I;UAGrB,IAAI,UAAU,kFAAV,CAAJ,C;YACI,mBAAY,gBAAZ,E
AAY,wBAAZ,YAAyB,S;;YAEzB,WAAW,I;;QAGnB,UAAU,mBAAY,OAAZ,C;QAEV,mBAAc,CAAd,YAAsB,I
AAtB,Y;UACI,gBAAc,mBAAY,OAAZ,C;UACd,mBAAY,OAAZ,IAAqB,I;UAGrB,IAAI,UAAU,kFAAV,CAAJ,
C;YACI,mBAAY,OAAZ,IAAuB,S;YACvB,UAAU,mBAAY,OAAZ,C;;YAEV,WAAW,I;;;;MAIvB,IAAI,QAAJ,C
;QACI,YAAO,mBAAY,UAAU,WAAV,IAAZ,C;MAEX,OAAO,Q;K;iCAGX,Y;MACI,WA7agD,mBAAY,cA6an
C,SA7amC,IAAZ,C;MA8ahD,IAAI,cAAO,IAAX,C;QACgB,OAAZ,kBAAY,EAAK,IAAL,EAAW,WAAX,EAAiB
,IAAjB,C;;QACT,IxKtS6C,CAAC,cwKsS9C,C;UACS,OAAZ,kBAAY,EAAK,IAAL,EAAW,WAAX,EAAiB,kBA
AY,OAA7B,C;UACA,OAAZ,kBAAY,EAAK,IAAL,EAAW,CAAX,EAAc,IAAd,C;;;MAEhB,cAAO,C;MACP,YA
AO,C;K;2CAGX,iB;MAGe,IAAC,IAAD,EAcJ,M;MAfP,WACW,eAAC,OAAI,KAAM,OAAN,IAAc,SAAlB,GAA
wB,KAAxB,GAAmC,aAAa,KAAb,EAAoB,SAApB,CAApC,uB;MAEX,WA7bgD,mBAAY,cA6bnC,SA7bmC,IA
AZ,C;MA8bhD,IAAI,cAAO,IAAX,C;Q/J2XJ,U+J1XQ,kB/J0XR,E+J1X6B,I/J0X7B,EAD+F,CAC/F,E+J1XgD,W/
J0XhD,E+J1XiE,I/J0XjE,C;;Q+JzXW,IxKtT6C,CAAC,cwKsT9C,C;U/JyXX,U+JxXQ,kB/JwXR,E+JxX6B,I/JwX7
B,E+JxXuD,C/JwXvD,E+JxXuE,W/JwXvE,E+JxXwF,kBAAY,O/JwXpG,C;UAAA,U+JvXQ,kB/JuXR,E+JvX6B,I
/JuX7B,E+JvXuD,kBAAY,OAAZ,GAAmB,WAAnB,I/JuXvD,E+JvX6F,C/JuX7F,E+JvX2G,I/JuX3G,C;;;M+JrXI,I
AAI,IAAK,OAAL,GAAY,SAAhB,C;QACI,KAAK,SAAL,IAAa,I;;MAIjB,OAAO,qD;K;mCAGX,Y;MAEI,OAAO
,qBAAQ,gBAAmB,SAAnB,OAAR,C;K;+CAGX,iB;MAC0D,4BAAQ,KAAR,C;K;+CAC1D,Y;MAA0C,qB;K;IAE
1C,gC;MAAA,oC;MACI,0BrHriBuC,E;MqHsiBvC,sBAAiC,U;MACjC,4BAAuC,E;K;yDAEvC,oC;MAEI,kBAAk
B,eAAe,eAAgB,CAA/B,K;MAClB,IAAI,eAAc,WAAd,QAA4B,CAAhC,C;QACI,cAAc,W;MAClB,IAAI,eAAc,U
AAd,QAA6B,CAAjC,C;QACI,cAAkB,cAAc,UAAlB,GAAgC,UAAhC,GAAmD,U;MACrE,OAAO,W;K;;;IAZf,4
C;MAAA,2C;QAAA,0B;;MAAA,oC;K;qDAgBA,qB;MAEI,WAvegD,mBAAY,cAuenC,SAvemC,IAAZ,C;MAwe
hD,WAAe,kBAAa,cAAO,IAAxB,GAA8B,WAA9B,GAAwC,cAAO,kBAAY,OAAnB,I;MACnD,UAAU,IAAV,EA
AgB,cAAhB,C;K;;IA5iBJ,iD;MAAA,oD;MAGwC,+B;MApB5C,sB;MAqBsB,Q;MACV,wBAAmB,CAAnB,C;QA
AwB,4D;WACxB,sBAAkB,CAAlB,C;QAAuB,uBAAa,eAAb,O;;QACf,MAAM,gCAAyB,uBAAoB,eAA7C,C;MA
HlB,0B;MAJJ,Y;K;IAWA,kC;MAAA,oD;MAGoB,+B;MA/BxB,sB;MAgCQ,sBAAc,qD;MAJlB,Y;K;IAOA,4C;M
AAA,oD;MAG2C,+B;MAtC/C,sB;MAuCQ,sBzJrB8D,YyJqBhD,QzJrBgD,C;MyJsB9D,aAAO,mBAAY,O;MACn
B,IAAI,mB9K+qPD,YAAQ,C8K/qPX,C;QAA2B,sBAAc,qD;MAN7C,Y;K;IC5BJ,4B;MAMoB,Q;M/KghqBA,U;
MADhB,UAAe,C;MACf,uD;QAAgB,cAAhB,iB;QACI,YAAgB,O+KlhqBiB,O/KkhqBjC,I;;M+KlhqBJ,aAAa,iB/K
ohqBN,G+KphqBM,C;MACb,wBAAgB,SAAhB,gB;QAAgB,gBAAA,SAAhB,M;QACW,SAAP,MAAO,EAAO,S
AAP,C;;MAEX,OAAO,M;K;IAGX,0B;MASiB,Q;MAFb,YAAY,iBAAa,gBAAb,C;MACZ,YAAY,iBAAa,gBAAb,
C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,KAAM,WAAI,IAAK,MAAT,C;QACN,KAAM,WAA
I,IAAK,OAAT,C;;MAEV,OAAO,UAAS,KAAT,C;K;gGAGX,qB;MAWW,4B;MAAA,U;QAAAqB,OAAL,S/K0qPh
B,YAAQ,C;;M+K1qPf,W;K;oFAGJ,mC;MAUI,O/K6pPO,qBAAQ,C+K7pPf,GAAe,cAAf,GAAAmC,S;K;IAGvC,iD
;MAMI,IAAI,cAAS,KAAb,C;QAAAoB,OAAO,I;MAC3B,IAAI,qBAAAgB,aAAAhB,IAAAiC,SAAK,OAAL,KAAAa,KA
AM,OAAAxD,C;QAA8D,OAAO,K;MAErE,4C;QACI,SAAS,UAAK,CAAL,C;QACT,SAAS,MAAM,CAAN,C;QA
ET,IAAI,OAAO,EAAX,C;UACI,Q;eACG,IAAI,cAAc,UAAlB,C;UACH,OAAO,K;;QAIP,0BAAsB,kBAAtB,C;U
AA4C,IAAI,CAAI,kBAAH,EAAG,EAAkB,EAAlB,CAAR,C;YAA+B,OAAO,K;eAClF,8BAAsB,sBAAtB,C;UAA
4C,IAAI,CAAI,cAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAC9E,+BAAsB,uBAAtB,C;UAA4C,IA
AI,CAAI,cAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAC9E,6BAAsB,qBAAtB,C;UAA4C,IAAI,CA
AI,cAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAC9E,8BAAsB,sBAAtB,C;UAA4C,IAAI,CAAI,cAA

H,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAC9E,+BAAsB,uBAAtB,C;UAA4C,IAAI,CAAI,cAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAC9E,gCAAsB,wBAAtB,C;UAA4C,IAAI,CAAI,cAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAC9E,8BAAsB,sBAAtB,C;UAA4C,IAAI,CAAI,cAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAC9E,iCAAsB,yBAAtB,C;UAA4C,IAAI,CAAI,cAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAE9E,qCAAsB,6BAAtB,C;UAA4C,IAAI,CAAI,gBAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAC9E,sCAAsB,8BAAtB,C;UAA4C,IAAI,CAAI,gBAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAC9E,oCAAsB,4BAAtB,C;UAA4C,IAAI,CAAI,gBAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAC9E,qCAAsB,6BAAtB,C;UAA4C,IAAI,CAAI,gBAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;eAAE,IAAI,YAAM,EAAN,CAAJ,C;UAAc,OAAO,K;;MAIrC,OAAO,I;K;IAGX,4C;MAKI,K;IAAkB,OAAO,M;MAAzB,aAAa,CAAK,eAAL,EAAK,gBAAK,EAAL,CAAK,eAAL,EAAK,CAAK,eAAL,EAAK,QAAA,iD,CAAA,kBAAA,I;MAAA,C;IAAA,C;IAAA,C;I

AKI,OAAI,QAAS,OAAT,KAAiB,CAArB,GAAwB,gBAAxB,GAAyC,iBAAU,sBAAkB,QAAlB,EAAwC,IAAxC,
CAAV,C;K;IAE7C,iC;MAKI,OAAI,QAAS,OAAT,KAAiB,CAArB,GAAwB,gBAAxB,GAAyC,iBAAU,sBAAkB,
QAAlB,EAAwC,IAAxC,CAAV,C;K;IAE7C,gC;MAI2D,OAAI,eAAJ,GAAqB,OAAO,OAAP,CAArB,GAA0C,W;
K;IAErG,mC;MAImE,OAAS,cAAT,QAAS,C;K;gFAE5E,yB;MAaA,gE;MAbA,6B;QAyBI,WAAW,eAAduE,IAAcvE,C;QWCX,iBAAAc,CAAAd,UXfkF,IWelF,U;UXA6B,eAf2D,IAAevD,CWCtB,KXDsB,CAAJ,C;;QAfyC,OAgB/D,I;O;
KA3BX,C;8FAaA,yB;MAAA,gE;MAAA,6B;QAYI,WAAW,eAAA,a,IAAAb,C;QWCX,iBAAAc,CAAAd,UXAO,IWAP,
U;UXA6B,eAAI,KWCtB,KXDsB,CAAJ,C;;QAC7B,OAAO,I;O;KAdX,C;wFAiBA,yB;Me1FA,+D;Mf0FA,gC;Qet
F0B,gBAAf,gB;QfsGkB,aW3FzB,W;QX2FA,OW1FO,SIZoC,Q;O;KfsF/C,C;yFAwBA,yB;Me3GA,4E;MAAA,gE;
Mf2GA,0C;QevGI,qBf2HyB,Qe3HzB,C;QAC8B,gBAAvB,ef0HkB,Qe1HlB,C;Qf0H4B,aWvHnC,W;QXuHA,OWt
HO,SIJ4C,Q;O;KfsGvD,C;IAiCI,mC;MAAQ,uBAAG,iBAAO,CAAP,IAAH,C;K;IAQR,qC;MAAQ,OAAA,SAAK,
KAAL,GAAY,CAAZ,I;K;4FAEZ,qB;MAK4D,QAAC,mB;K;kGAE7D,qB;MAWI,OAAO,qBAAgB,SAAK,U;K;sF
AGhC,yB;MAAA,qD;MAAA,4B;QAKgE,uCAAQ,W;O;KALxE,C;sFAOA,yB;MAAA,qD;MAAA,4B;QAKoD,uC
AAQ,W;O;KAL5D,C;sFAOA,mC;MASI,OAAI,mBAAJ,GAAAe,cAAf,GAAAmC,S;K;4FAGvC,+B;MAQoH,OAAA,
SAAK,qBAAY,QAAZ,C;K;IAGzH,uC;MAK+E,kBAAhB,0B;MAAwB,+B;MAAxB,OW5MpD,W;K;IX+MX,yC;
MAAkD,QAAM,cAAN,C;aAC9C,C;UAD8C,OACzC,W;aACL,C;UAF8C,OAEzC,OAAO,sBAAK,CAAL,CAAP,
C;;UAFyC,OAGtC,S;;K;IAGZ,8D;MAgBkE,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,c;MCjG,WAAW,c
AAX,EAAiB,SAAjB,EAA4B,EAA5B,EAAgC,OAAhC,EAAyC,C;MACV,aAAU,S;MACV,WAAW,UAAU,CAAV,I;MAEX,OAAO,OAA
O,IAAd,C;QACI,UAAW,GAAY,GAAN,IAAM,KAAK,C;QAC5B,2B,aAAa,sBAAI,GAAJ,C;QACb,UAAU,cAAc,M
AAd,EAAsB,OAAtB,C;QAEV,IAAI,MAAM,CAAV,C;UACI,MAAM,MAAM,CAAN,I;aACL,IAAI,MAAM,CAAV,C;UACI,MAAM,CAAN,I;;UAEJ,OAAO,G;;MAEf,OAAO,EAAE,MAAM,CAAN,IAAF,K;K;IAGX,4E
;MAe8E,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,c;MC7G,WAAW,cAAX,EAAiB,SAAjB,EAA4B,EAA5B,OAA
5B,C;MACV,aAAU,S;MACV,WAAW,UAAU,CAAV,I;MAEX,OAAO,OAAO,IAAd,C;QACI,UAAW,GAAY,GAAN,IAAM,KAAK,C;QAC5B,2B,aAAa,sBAAI,GAAJ,C;QACb,UAAU,cAAc,WAAd,EAA2B,SAA3B,C;QAEV,IAAI,MAAM,CAAV,C;UACI,MAAM,MAAM,CAAN,I;aACL,IAAI,MAAM,CAAV,C;UACI,MAAM,CAAN,I;;UAEJ,OAAO,G;;MAEf,OAAO,EAAE,MAAM,CAAN,IAAF,K;K;IAGX,kG
AGgB,yB;MAAA,8D;MAAA,4D;MAsBqC,8D;QAAA,qB;UAAA,iBAAe,EAAf,EAAmB,SAAnB,EAAAC,CAAAC,SAAAC,EAAAC,
CAAAC,SAAAC,EAAAC,CAAAC,SAAAC,EAAAC,CAAAC,SAAAC,EAAAC,

U,WAAY,GAAZ,C;UA6DuF,U;UAAjG,W1G2GZ,a0GvKgB,G1GuKhB,E0G3GiD,UA5DhB,GA4DgB,EA5DK,uBAAuB,CA4DjE,WA5D8E,mBAAY,GAAZ,CA4D1B,GAAW,qBA5D1C,GA4D0C,EA5DxB,CA4DwB,CAAX,GAA6C,UA5DvE,WA4DuE,6DAA5D,EA5DE,CA4DF,C1G2GjD,C;;Q0G5GA,OACY,W;O;KA7BhB,C;iFAgCA,yB;MAAA,gB;MAAA,8B;M1GhFA,wE;M0GgFA,qD;QA7FW,kB1GqBiD,oB;Q0GM9C,Q;QAAA,OAAK,0B;QAAf,OAAU,cAAV,C;UAAU,mB;UACN,UAAU,sBAAM,CAAN,C;UACV,kBAAkB,sBAAY,GAAZ,C;UAkFiD,U;UAjFnE,W1GuKJ,a0GvKgB,G1GuKhB,E0GtFgC,UAjFsB,uBAAuB,CAAC,WAAY,mBAAY,GAAZ,CAiFhD,kBAA6B,UAjFjC,WAiFiC,6DAAvC,EAjFmB,CAiFnB,C1GsFhC,C;;Q0GvFA,OA9EO,W;O;KA6DX,C;oFAoBA,yB;MAAA,gB;MAAA,8B;MAAA,kE;QAtFc,Q;QAAA,OAAK,0B;QAAf,OAAU,cAAV,C;UAAU,mB;UACN,UAAU,sBAAM,CAAN,C;UACV,kBA2GQ,WA3GU,GAAZ,C;UA2GgE,U;UA1G1E,W1G6DZ,a0GvKgB,G1GuKhB,E0G7D+C,UA1GO,uBAAuB,CA0GjE,WA1G8E,mBAAY,GAAZ,CA0GjC,kBAA6B,UA1GhD,WA0GgD,6DAAvC,EA1GE,CA0GF,C1G4DhC,C;;Q0G7DA,OACY,W;O;KAvBhB,C;qFA0BA,yB;MAAA,gB;MAAA,8B;M1GhFA,wE;M0GgFA,uC;QA3IW,kB1GqBiD,oB;Q0GM9C,Q;QAAA,OAAK,0B;QAAf,OAAU,cAAV,C;UAAU,mB;UACN,UAAU,sBAAM,CAAN,C;UACV,kBAAkB,sBAAY,GAAZ,C;UAkFiD,U;UAjFnE,WAiFiC,kBAAK,CAAL,C;UAjFjC,oB;UAiFiC,U;UAjFjC

AA,QAAI,K;K;4FACH,Y;MAAQ,OAAA,QAAI,O;K;6FACJ,Y;MAAQ,OAAA,QAAI,Q;K;8DAEvD,e;MAAmD,gBAAJ,Q;MAAI,4B;M5K4PxC,Q;MALX,YAAY,oB4KvPyD,G5KuPzD,C;MACZ,IAAI,iBAAiB,CAAC,4B4KxP+C,G5KwP/C,CAAtB,C;QACI,2B4KzPwE,mB;;Q5K4PxE,2BAAO,sE;;M4K5PoC,+B;K;;IAGN,mD;MAAC,wB;MAA2C,0B;K;4EAA3C,Y;MAAA,yB;K;iDAC1C,iB;MAA4C,OAAI,OAAJ,QAAI,EAAO,KAAP,C;K;mDAChD,Y;MA+B,OAAI,SAAJ,QAAI,C;K;mDACnC,Y;MAAkC,OAAA,QAAI,W;K;iGACf,Y;MAAQ,OAAA,QAAI,K;K;kDACnC,Y;MAAkC,OAAA,QAAI,U;K;4DACtC,e;MA4C,OAAA,QAAI,mBAAY,GAAZ,C;K;8DAChD,iB;MAgE,OAAA,QAAI,qBAAc,KAAd,C;K;oDACpE,e;MA+B,OAAA,QAAI,WAAI,GAAJ,C;K;iGACF,Y;MAAQ,OAAA,QAAI,K;K;mGACH,Y;MAAQ,OAAA,QAAI,O;K;oGACU,Y;MAAQ,OAAA,QAAI,Q;K;sDAE5E,sB;MAAyC,OAAA,QAAI,aAAI,GAAJ,EAAS,KAAT,C;K;uDAC7C,e;MAAkC,OAAA,QAAI,cAAO,GAAP,C;K;yDACtC,gB;MA2C,gBAAO,IAAP,C;K;gDAC/C,Y;MAAuB,QAAI,Q;K;qEAE3B,e;MAAmD,gBAAJ,Q;MAAI,4B;M5KuOxC,Q;MALX,YAAY,oB4KlOyD,G5KkOzD,C;MACZ,IAAI,iBAAiB,CAAC,4B4KnO+C,G5KmO/C,CAAtB,C;QACI,2B4KpOwE,mB;;Q5KuOxE,2BAAO,sE;;M4KvOoC,+B;K;;I5KuOnD,oB;MAAA,wB;MACI,8C;K;gCAEA,iB;MAA4C,oCAAsB,KAAA,M,U;K;kCACxE,Y;MAA+B,Q;K;kCAC/B,B,Y;MAAkC,W;K;gFAEX,Y;MAAQ,Q;K;iCAC/B,B,Y;MAAkC,W;K;2CAElC,e;MAA+C,C,Y;K;6CAC/C,iB;MAAsD,Y;K;mCACtD,e;MAAwC,W;K;mFACX,Y;MAAQ,6B;K;gFAC/B,B,Y;MAAQ,6B;K;kFACI,Y;MAAQ,8B;K;uCAEjD,Y;MAAiC,6B;K;;;IAIjB,rC,gC;MAAA,+B;QAAA,c;;MAAA,wB;K;IAoBA,oB,oB;MAMuE,Q;MAA7B,OAA6B,uE;K;IAEvE,wB;MAAI,I,OAAI,I,KAAI,M,EAAI,I,KAAI,M,EAAI,I,KAAI,M,E

AYA,yB;MAAA,0D;MAAA,uE;MAAA,uC;QAYW,kBAAU,mBAAoB,YAAY,cAAZ,CAApB,C;Qa+wBD,Q;QAAA,Obr0BT,iBaq0BS,W;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;Ubr0Ba,Was0Bb,abhxByC,SagxBrC,CAAY,OAAZ,CAAJ,EAAyC,Obt0BC,Mas0B1C,C;;QbhxBhB,OAtD6B,W;O;KA0CjC,C;0FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3FsE,QAAQ,W;QA2F5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;8FAgBA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3GsE,QAAQ,W;QA2G5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,MAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;yFAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3HsE,QAAQ,W;QA2H5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3IsE,QAAQ,W;QA2I5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3JsE,QAAQ,W;QA2J5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3KsE,QAAQ,W;QA2K5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3LsE,QAAQ,W;QA2L5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3MsE,QAAQ,W;QA2M5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3NsE,QAAQ,W;QA2N5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3OsE,QAAQ,W;QA2O5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3PsE,QAAQ,W;QA2P5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3QsE,QAAQ,W;QA2Q5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3RsE,QAAQ,W;QA2R5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3SsE,QAAQ,W;QA2S5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3TsE,QAAQ,W;QA2T5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3UsE,QAAQ,W;QA2U5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3VsE,QAAQ,W;QA2V5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3WsE,QAAQ,W;QA2W5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3XsE,QAAQ,W;QA2X5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3YsE,QAAQ,W;QA2Y5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3ZsE,QAAQ,W;QA2Z5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3asE,QAAQ,W;QA2a5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3bsE,QAAQ,W;QA2b5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3csE,QAAQ,W;QA2c5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3dsE,QAAQ,W;QA2d5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3esE,QAAQ,W;QA2e5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3fsE,QAAQ,W;QA2f5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3gBsE,QAAQ,W;QA2gB5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3hBsE,QAAQ,W;QA2hB5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3isE,QAAQ,W;QA2i5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3jsE,QAAQ,W;QA2j5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3ksE,QAAQ,W;QA2k5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3lsE,QAAQ,W;QA2l5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3msE,QAAQ,W;QA2m5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3nsE,QAAQ,W;QA2n5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3osE,QAAQ,W;QA2o5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3psE,QAAQ,W;QA2p5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3qsE,QAAQ,W;QA2q5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3rsE,QAAQ,W;QA2r5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3ssE,QAAQ,W;QA2s5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3tsE,QAAQ,W;QA2t5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3usE,QAAQ,W;QA2u5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3vsE,QAAQ,W;QA2v5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3wsE,QAAQ,W;QA2w5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3xsE,QAAQ,W;QA2x5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3ysE,QAAQ,W;QA2y5F,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,hB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,EAAAe,KAAM,MAArB,rB,C;;QAGf,OAAO,M;O;KAbX,C;+FAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3zsE,QAAQ,W;QA2z

AAK,EAAO,QAAP,C;O;KALT,C;8FAQA,8B;MAKI,SAAK,cAAO,OAAP,C;K;8FAGT,yB;MAAA,sD;MAAA,sC;QAKS,UAAL,SAAK,EAAU,QAAV,C;O;KALT,C;8FAQA,yB;MAAA,sD;MAAA,sC;QAKS,UAAL,SAAK,EAAU,QAAV,C;O;KALT,C;8FAQA,yB;MAAA,sD;MAAA,sC;QAKS,UAAL,SAAK,EAAU,QAAV,C;O;KALT,C;IAQA,qC;MAIU,IAIe,I;MAHjB,kBADE,QACF,c;QAAiB,OAAO,yBAAO,QAAP,C;;QAEpB,aAAsB,K;QACT,0B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,IAAI,oBAAI,IAAJ,CAAJ,C;YAAe,SAAS,I;;QAC5B,5B,OAAO,M;;K;IAKnB,uC;MAKiB,Q;MADb,aAAsB,K;MACT,0B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAI,oBAAI,IAAJ,CAAJ,C;UAAe,SAAS,I;;MAE5B,E5B,OAAO,M;K;IAGX,uC;MAII,OAAO,yBAAgB,B,OAAT,QAAS,CAAhB,B,C;K;IAGX,0C;MAIW,iBAAmB,gCAAT,QAAS,EAAgC,SAAhC,CAAhB,B,C;K;IAGX,0C;MAIW,iBAAmB,gCAAT,QAAS,EAAgC,SAAhC,CAAhB,B,C;K;IAIqH,/D,0C;MAII,UAAmB,8BAAT,QAAS,C;MACnB,O9K0EwD,C8K1EjD,G9K0EkD,U8K1ElD,IAAoB,4BAAU,GAAV,C;K;IAG/B,0C;MAII,OpLqoPO,EoLroPA,QpL6jPA,YAAQ,CAAwE,CoLroPA,IAAyB,4BAAmB,8BAAT,QAAS,C;K;IAGpC,0C;MAIW,iBAAmB,gCAAT,QAAS,EAAgC,SAAhC,CAAhB,B,C;MAIqH,Q;MA6H7B,OA7H2D,CAA9B,sEAAU,GAAV,C;K;IAIgI

B,C;UACI,eAAQ,C;UACR,aACa,mF;UACb,mBAAY,I;UACZ,OAAO,M;;UAEH,MAAM,yB;;K;uDAItB,Y;MACI,IAAI,CAAC,cAAL,C;QAAgB,MAAM,6B;;QAA8B,OAAO,W;K;2DAG/D,Y;MAA4C,QAAM,YAAN,C;aACxC,C;UADwC,OAC1B,6B;aACd,C;UAFwC,OAExB,6BAAsB,sBAAtB,C;;UAFwB,OAGhC,6BAAsB,uCAAoC,YAA1D,C;;K;IAOqC,4E;MAAA,oB;QACzC,wCAAW,C;QAAX,OACA,yB;O;K;oDALR,+B;MACI,mBAAY,K;MACZ,eAAQ,C;MACR,OAA6C,0CAAtC,c;K;IAUsC,+E;MAAA,oB;QACzC,wCAAW,C;QAAX,OACA,yB;O;K;yDANR,kC;MACI,IAAI,CAAC,QAAS,UAAd,C;QAAyB,M;MACzB,sBAAe,Q;MACf,eAAQ,C;MACR,OAA6C,6CAAtC,c;K;2DAMX,kB;M1HjBO,Q;MADP,e0HoBI,M1HpBJ,C;MACO,Q0HmBH,M1HnBG,+D;M0HoBH,eAAQ,C;K;kGAIR,Y;MAAQ,0C;K;;IzK1LhB,oD;MAQuF,wC;K;IARvF,8CASI,Y;MAAuC,8B;K;IAT3C,gF;sFAAA,yB;MAAA,kC;MAAA,0C;MAAA,kD;QAQuF,wC;O;MARvF,4CASI,Y;QAAuC,8B;O;MAT3C,8E;MAAA,2B;QAQuF,2C;O;KARvF,C;IAiBgE,+C;MAAA,mB;QAAE,sB;O;K;IALlE,kC;MAKuD,OAAkB,2CAAT,+BAAS,E;K;IAEzE,8B;MAK6D,OAAI,Qb2rPtD,YAAQ,Ca3rP0C,GAAwB,eAAxB,GAAsD,WAAT,QAAS,C;K;IAEnH,yB;MAG8C,kC;K;IAE9C,yB;MAAA,6B;K;uCACI,Y;MAA6C,kC;K;2CAC7C,a;MAA4B,kC;K;2CAC5B,a;MAA4B,kC;K;;;IAHhC,qC;MAAA,oC;QAAA,mB;;MAAA,6B;K;oFAMA,yB;MAAA,2D;MAAA,4B;QAAM,uCAAQ,e;O;KANpE,C;IAgBgC,4F,mH;MAAA,wC;MAAA,6B;MAAA,yB;MAAA,wC;MAAA,wD;MAAA,kC;K;;;;kDAAA,Y;;;;;cACxF,eAAe,uBAAa,W;cAC5C,5B,IAAI,QAAS,UAAb,C;gBACI,gB;gCAAA,sCAAS,QAAT,QAAA,O;oBAAA,2C;yBAAA,yB;gBAAA,Q;;gBAEA,gB;gCAAA,sCAAS,iCAAT,O;oBAAA,2C;yBAAA,yB;gBAAA,Q;;;;;;cAAJ,AJJ,W;;cAAA,W;;;;;;;;;;;;;;;;K;IADwF,gF;MAAA,yD;uBAAA,uG;YAAA,S;iBAAA,Q;;iBAAA,Q;;O;KAAA,C;IAAA

CvB,IAAI,gBAAiB,UAArB,C;YACI,oBAAe,gB;YACf,OAAO,I;;;;MAInB,OAAO,I;K;;4CA9Bf,Y;MAAuC,0D;K;;IAoC9B,6I;MAAA,wC;MAAA,6B;MAAA,yB;MAAA,4C;MAAA,kD;MAAA,gD;MAAA,wB;MAAA,yB;MAAA,kC;K;;;;yDAAA,Y;;;;;kBAGyC,I;iCAFlC,C;cACI,sD;cAAhB,gB;;;;;cAAA,KAAgB,yBAAhB,C;gBAAA,gB;;;;cAAgB,oC;cACZ,aAAa,6BAAU,oBAAmB,uBAAnB,EAAmB,+BAAnB,QAAV,EAAuC,OAAvC,C;cACb,gB;8BAAA,sCAAS,4BAAS,MAAT,CAAS,O;kBAAA,2C;uBAAA,yB;cAAA,Q;;cAFJ,gB;;;cAIJ,W;;;;;;;;;;;;;;;K;IANS,0F;MAAA,yD;uBAAA,iI;YAAA,S;iBAAA,Q;;iBAAA,uB;O;K;IADb,wD;MACI,gBAAS,kDAAT,C;K;;;;IAoByB,qD;MACzB,0B;MACA,8B;MACA,0B;MC3TA,IAAI,ED+TQ,qBAAc,CC/Tb,TtB,CAAJ,C;QACI,cD8T2B,+CAA4C,iB;QC7TvE,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MAFV,IAAI,EDgUQ,mBAAY,CChUb,UhB,CAAJ,C;QACI,gBD+T+B,yBAAsB,6C;QC9T3D,eAAM,gCAAyB,SAAQ,WAAjC,C,C;;MAFV,IAAI,EDiUQ,mBAAY,CCjUb,UjB,CAAJ,C;QACI,iBCjUpB,UhB,CAAJ,C;QACI,gBDgUgC,yBAAsB,C;QC/U1D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDkVQ,mBAAY,CClVb,UjB,CAAJ,C;QACI,iBCkVpB,UhB,CAAJ,C;QACI,gBDiVgC,yBAAsB,C;QChW1D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDmWQ,mBAAY,CCnWb,UjB,CAAJ,C;QACI,iBCmWpB,UhB,CAAJ,C;QACI,gBDkWgC,yBAAsB,C;QCjX1D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDoXQ,mBAAY,CCpXb,UjB,CAAJ,C;QACI,iBCoXpB,UhB,CAAJ,C;QACI,gBDmXgC,yBAAsB,C;QClY1D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDqYQ,mBAAY,CCrYb,UjB,CAAJ,C;QACI,iBCqYpB,UhB,CAAJ,C;QACI,gBDoYgC,yBAAsB,C;QCnZ1D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDsZQ,mBAAY,CCtZb,UjB,CAAJ,C;QACI,iBCsZpB,UhB,CAAJ,C;QACI,gBDqZgC,yBAAsB,C;QCpa1D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDuaQ,mBAAY,CCvab,UjB,CAAJ,C;QACI,iBCuapB,UhB,CAAJ,C;QACI,gBDsagC,yBAAsB,C;QCrb1D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDwbQ,mBAAY,CCxbb,UjB,CAAJ,C;QACI,iBCwbpB,UhB,CAAJ,C;QACI,gBDubgC,yBAAsB,C;QCtc1D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDycQ,mBAAY,CCzcb,UjB,CAAJ,C;QACI,iBCycpB,UhB,CAAJ,C;QACI,gBDwcgC,yBAAsB,C;QCvd1D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDydQ,mBAAY,CCzdb,UjB,CAAJ,C;QACI,iBCydpB,UhB,CAAJ,C;QACI,gBDwdgC,yBAAsB,C;QCve1D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyeQ,mBAAY,CCzeb,UjB,CAAJ,C;QACI,iBCyepB,UhB,CAAJ,C;QACI,gBDweiC,yBAAsB,C;QCvf3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyfQ,mBAAY,CCzfb,UjB,CAAJ,C;QACI,iBCyfpB,UhB,CAAJ,C;QACI,gBDwfiC,yBAAsB,C;QCvg3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyhBQ,mBAAY,CCzhBb,UjB,CAAJ,C;QACI,iBCyhBpB,UhB,CAAJ,C;QACI,gBDwhBiC,yBAAsB,C;QCvi3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyiBQ,mBAAY,CCziBb,UjB,CAAJ,C;QACI,iBCyiBpB,UhB,CAAJ,C;QACI,gBDwiBiC,yBAAsB,C;QCvj3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyjBQ,mBAAY,CCzjBb,UjB,CAAJ,C;QACI,iBCyjBpB,UhB,CAAJ,C;QACI,gBDwjBiC,yBAAsB,C;QCvk3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDykBQ,mBAAY,CCzkBb,UjB,CAAJ,C;QACI,iBCykBpB,UhB,CAAJ,C;QACI,gBDwkBiC,yBAAsB,C;QCvl3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDylBQ,mBAAY,CCzlBb,UjB,CAAJ,C;QACI,iBCylBpB,UhB,CAAJ,C;QACI,gBDwlBiC,yBAAsB,C;QCvm3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDymBQ,mBAAY,CCzmBb,UjB,CAAJ,C;QACI,iBCymBpB,UhB,CAAJ,C;QACI,gBDwmBiC,yBAAsB,C;QCvn3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDynBQ,mBAAY,CCznBb,UjB,CAAJ,C;QACI,iBCynBpB,UhB,CAAJ,C;QACI,gBDwnBiC,yBAAsB,C;QCvo3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyoBQ,mBAAY,CCzoBb,UjB,CAAJ,C;QACI,iBCyoBpB,UhB,CAAJ,C;QACI,gBDwoBiC,yBAAsB,C;QCvp3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDypBQ,mBAAY,CCzpBb,UjB,CAAJ,C;QACI,iBCypBpB,UhB,CAAJ,C;QACI,gBDwpBiC,yBAAsB,C;QCvq3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyqBQ,mBAAY,CCzqBb,UjB,CAAJ,C;QACI,iBCyqBpB,UhB,CAAJ,C;QACI,gBDwqBiC,yBAAsB,C;QCvr3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyrBQ,mBAAY,CCzrBb,UjB,CAAJ,C;QACI,iBCyrBpB,UhB,CAAJ,C;QACI,gBDwrBiC,yBAAsB,C;QCvs3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDysBQ,mBAAY,CCzsBb,UjB,CAAJ,C;QACI,iBCysBpB,UhB,CAAJ,C;QACI,gBDwsBiC,yBAAsB,C;QCvt3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDytBQ,mBAAY,CCztBb,UjB,CAAJ,C;QACI,iBCytBpB,UhB,CAAJ,C;QACI,gBDwtBiC,yBAAsB,C;QCvu3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyuBQ,mBAAY,CCzuBb,UjB,CAAJ,C;QACI,iBCyuBpB,UhB,CAAJ,C;QACI,gBDwuBiC,yBAAsB,C;QCvv3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyvBQ,mBAAY,CCzvBb,UjB,CAAJ,C;QACI,iBCyvBpB,UhB,CAAJ,C;QACI,gBDwvBiC,yBAAsB,C;QCvw3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDywBQ,mBAAY,CCzwBb,UjB,CAAJ,C;QACI,iBCywBpB,UhB,CAAJ,C;QACI,gBDwwBiC,yBAAsB,C;QCvx3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyxBQ,mBAAY,CCzxBb,UjB,CAAJ,C;QACI,iBCyxBpB,UhB,CAAJ,C;QACI,gBDwxBiC,yBAAsB,C;QCvy3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyyBQ,mBAAY,CCzyBb,UjB,CAAJ,C;QACI,iBCyyBpB,UhB,CAAJ,C;QACI,gBDwyBiC,yBAAsB,C;QCvz3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,EDyzBQ,mBAAY,CCzzBb,UjB,CAAJ,C;QACI,iBCyzBpB,UhB,CAAJ,C;QACI,gBDwzBiC,yBAAsB,C;QCv03D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED01BQ,mBAAY,CC11Bb,UjB,CAAJ,C;QACI,iBC01BpB,UhB,CAAJ,C;QACI,gBDy1BiC,yBAAsB,C;QCx23D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED02BQ,mBAAY,CC12Bb,UjB,CAAJ,C;QACI,iBC02BpB,UhB,CAAJ,C;QACI,gBDy2BiC,yBAAsB,C;QCx33D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED03BQ,mBAAY,CC13Bb,UjB,CAAJ,C;QACI,iBC03BpB,UhB,CAAJ,C;QACI,gBDy3BiC,yBAAsB,C;QCx43D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED04BQ,mBAAY,CC14Bb,UjB,CAAJ,C;QACI,iBC04BpB,UhB,CAAJ,C;QACI,gBDy4BiC,yBAAsB,C;QCx53D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED06BQ,mBAAY,CC16Bb,UjB,CAAJ,C;QACI,iBC06BpB,UhB,CAAJ,C;QACI,gBDy6BiC,yBAAsB,C;QCx73D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED07BQ,mBAAY,CC17Bb,UjB,CAAJ,C;QACI,iBC07BpB,UhB,CAAJ,C;QACI,gBDy7BiC,yBAAsB,C;QCx83D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED08BQ,mBAAY,CC18Bb,UjB,CAAJ,C;QACI,iBC08BpB,UhB,CAAJ,C;QACI,gBDy8BiC,yBAAsB,C;QCx93D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED09BQ,mBAAY,CC19Bb,UjB,CAAJ,C;QACI,iBC09BpB,UhB,CAAJ,C;QACI,gBDy9BiC,yBAAsB,C;QCx+3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0+BQ,mBAAY,CC1+Bb,UjB,CAAJ,C;QACI,iBC0+BpB,UhB,CAAJ,C;QACI,gBDy+BiC,yBAAsB,C;QCx/3D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0/BQ,mBAAY,CC1/Bb,UjB,CAAJ,C;QACI,iBC0/BpB,UhB,CAAJ,C;QACI,gBDy/BiC,yBAAsB,C;QCxg4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0gCQ,mBAAY,CC1gCb,UjB,CAAJ,C;QACI,iBC0gCpB,UhB,CAAJ,C;QACI,gBDygCiC,yBAAsB,C;QCxh4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0hCQ,mBAAY,CC1hCb,UjB,CAAJ,C;QACI,iBC0hCpB,UhB,CAAJ,C;QACI,gBDyhCiC,yBAAsB,C;QCxi4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0iCQ,mBAAY,CC1iCb,UjB,CAAJ,C;QACI,iBC0iCpB,UhB,CAAJ,C;QACI,gBDyiCiC,yBAAsB,C;QCxj4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0jCQ,mBAAY,CC1jCb,UjB,CAAJ,C;QACI,iBC0jCpB,UhB,CAAJ,C;QACI,gBDyjCiC,yBAAsB,C;QCxk4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0kCQ,mBAAY,CC1kCb,UjB,CAAJ,C;QACI,iBC0kCpB,UhB,CAAJ,C;QACI,gBDykCiC,yBAAsB,C;QCxl4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0lCQ,mBAAY,CC1lCb,UjB,CAAJ,C;QACI,iBC0lCpB,UhB,CAAJ,C;QACI,gBDylCiC,yBAAsB,C;QCxm4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0mCQ,mBAAY,CC1mCb,UjB,CAAJ,C;QACI,iBC0mCpB,UhB,CAAJ,C;QACI,gBDymCiC,yBAAsB,C;QCxn4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0nCQ,mBAAY,CC1nCb,UjB,CAAJ,C;QACI,iBC0nCpB,UhB,CAAJ,C;QACI,gBDynCiC,yBAAsB,C;QCxo4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0oCQ,mBAAY,CC1oCb,UjB,CAAJ,C;QACI,iBC0oCpB,UhB,CAAJ,C;QACI,gBDyoCiC,yBAAsB,C;QCxp4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0pCQ,mBAAY,CC1pCb,UjB,CAAJ,C;QACI,iBC0pCpB,UhB,CAAJ,C;QACI,gBDypCiC,yBAAsB,C;QCxq4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0qCQ,mBAAY,CC1qCb,UjB,CAAJ,C;QACI,iBC0qCpB,UhB,CAAJ,C;QACI,gBDyqCiC,yBAAsB,C;QCxr4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0rCQ,mBAAY,CC1rCb,UjB,CAAJ,C;QACI,iBC0rCpB,UhB,CAAJ,C;QACI,gBDyrCiC,yBAAsB,C;QCxs4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0sCQ,mBAAY,CC1sCb,UjB,CAAJ,C;QACI,iBC0sCpB,UhB,CAAJ,C;QACI,gBDysCiC,yBAAsB,C;QCxt4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0tCQ,mBAAY,CC1tCb,UjB,CAAJ,C;QACI,iBC0tCpB,UhB,CAAJ,C;QACI,gBDytCiC,yBAAsB,C;QCxu4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0uCQ,mBAAY,CC1uCb,UjB,CAAJ,C;QACI,iBC0uCpB,UhB,CAAJ,C;QACI,gBDyuCiC,yBAAsB,C;QCxv4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0vCQ,mBAAY,CC1vCb,UjB,CAAJ,C;QACI,iBC0vCpB,UhB,CAAJ,C;QACI,gBDyvCiC,yBAAsB,C;QCxw4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0wCQ,mBAAY,CC1wCb,UjB,CAAJ,C;QACI,iBC0wCpB,UhB,CAAJ,C;QACI,gBDywCiC,yBAAsB,C;QCxx4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0xCQ,mBAAY,CC1xCb,UjB,CAAJ,C;QACI,iBC0xCpB,UhB,CAAJ,C;QACI,gBDyxCiC,yBAAsB,C;QCxy4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0yCQ,mBAAY,CC1yCb,UjB,CAAJ,C;QACI,iBC0yCpB,UhB,CAAJ,C;QACI,gBDyyCiC,yBAAsB,C;QCxz4D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED0zCQ,mBAAY,CC1zCb,UjB,CAAJ,C;QACI,iBC0zCpB,UhB,CAAJ,C;QACI,gBDyzCiC,yBAAsB,C;QCx05D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED21CQ,mBAAY,CC31Cb,UjB,CAAJ,C;QACI,iBC21CpB,UhB,CAAJ,C;QACI,gBD01CiC,yBAAsB,C;QCz25D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED22CQ,mBAAY,CC32Cb,UjB,CAAJ,C;QACI,iBC22CpB,UhB,CAAJ,C;QACI,gBD02CiC,yBAAsB,C;QCz35D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED23CQ,mBAAY,CC33Cb,UjB,CAAJ,C;QACI,iBC23CpB,UhB,CAAJ,C;QACI,gBD03CiC,yBAAsB,C;QCz45D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz55D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz65D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz75D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz85D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz95D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz+5D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz/5D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzg6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzh6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzi6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzj6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzk6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzl6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzm6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzn6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzo6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzp6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzq6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzr6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzs6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzt6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzu6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzv6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzw6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzx6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzy6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzz6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz06D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz16D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz26D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz36D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz46D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz56D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz66D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz76D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz86D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz96D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz+6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCz/6D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzg7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzh7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzi7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzj7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzk7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzl7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzm7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzn7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzo7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzp7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzq7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzr7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzs7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzt7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzu7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBAAsB,C;QCzv7D,gBAAM,gCAAyB,SAAQ,WAAjC,C,C;;K;sFAFV,IAAI,ED43CQ,mBAAY,CC53Cb,UjB,CAAJ,C;QACI,iBC43CpB,UhB,CAAJ,C;QACI,gBD23CiC,yBA

qB;QAAE,6B;O;K;IAX7C,wC;MAWI,OAA2D,cAApD,sBAAkB,YAAlB,EAAgC,qCAAhC,CAAoD,C;K;IAqBrC,iD;MAAA,mB;QAAE,mB;O;K;IAlB5B,gD;MAeI,OAAI,YAAJ,GACI,2BADJ,GAGI,sBAAkB,+BAAlB,EAA4B,YAA5B,C;K;IAER,wD;MAcI,6BAAkB,YAAlB,EAAgC,YAAhC,C;K;ILxpBJ,oB;MAAA,wB;MACI,8C;K;gCAEA,iB;MAA4C,oCAAmB,KAAM,U;K;kCACrE,Y;MA+B,Q;K;kCAC/B,Y;MAAkC,W;K;gFAEX,Y;MAAQ,Q;K;iCAC/B,Y;MAAkC,W;K;wCAClC,mB;MAAmD,Y;K;6CACnD,oB;MAAmE,OAAA,QAAS,U;K;kCAE5E,Y;MAA6C,kC;K;uCAE7C,Y;MAAiC,6B;K;;;IAdrC,gC;MAAA,+B;QAAA,c;;MAAA,wB;K;IAkBA,oB;MAIoC,6B;K;IAEpC,2B;MAMmD,OAAI,QAAS,OAAT,GAAgB,CAApB,GAAgC,MAAT,QAAS,CAAhC,C;K;uFAEhG,yB;MAAA,mD;MAAA,mB;QAKwC,iB;O;KALxC,C;6FAOA,yB;MAAA,uE;MAAA,mB;QAQsD,2B;O;KARtD,C;IAUA,kC;MAKiE,OAAS,aAAT,QAAS,EAAa,qBAAb,c,YAAY,QAAS,OAArB,rB,CAAqB,aAArB,EAAqB,CAArB,C;K;uFAE1E,yB;MAAA,2D;MAAA,mB;QAGgD,iB;O;KAHhD,C;IAKA,+B;MAAC2D,OAAS,aAAT,QAAS,EAAa,eAAb,EAAa,eAAb,Q,YAAY,QAAS,OAArB,rB,CAAqB,aAArB,EAAqB,CAArB,C;K;IAE5C,+B;MAAC,2D;MAAA,mB;QAAA,gD,qB;O;KAAA,C;IAAA,+B;MAAA,+B;QAAA,+B;UAAA,+B;YAAA,+B;cAAA,+B;gBAAA,+B;kBAAA,+B;oBAAA,+B;sBAAA,+B;wBAAA,+B;0BAAA,+B;4BAAA,+B;8BAAA,+B;gCAAA,+B;kCAAA,+B;oCAAA,+B;sCAAA,+B;wCAAA,+B;0CAAA,+B;4CAAA,+B;8CAAA,+B;gDAAA,+B;kDAAA,+B;oDAAA,+B;sDAAA,+B;wDAAA,+B;0DAAA,+B;4DAAA,+B;8DAAA,+B;gEAAA,+B;kEAAA,+B;oEAAA,+B;sEAAA,+B;wEAAA,+B;0EAAA,+B;4EAAA,+B;8EAAA,+B;gFAAA,+B;kFAAA,+B;oFAAA,+B;sFAAA,+B;wFAAA,+B;0FAAA,+B;4FAAA,+B;8FAAA,+B;gGAAA,+B;kGAAA,+B;oGAAA,+B;sGAAA,+B;wGAAA,+B;0GAAA,+B;4GAAA,+B;8GAAA,+B;gHAAA,+B;kHAAA,+B;oHAAA,+B;sHAAA,+B;wHAAA,+B;0HAAA,+B;4HAAA,+B;8HAAA,+B;gIAAA,+B;kIAAA,+B;oIAAA,+B;sIAAA,+B;wIAAA,+B;0IAAA,+B;4IAAA,+B;8IAAA,+B;gJAAA,+B;kJAAA,+B;oJAAA,+B;sJAAA,+B;wJAAA,+B;0JAAA,+B;4JAAA,+B;8JAAA,+B;gKAAA,+B;kKAAA,+B;oKAAA,+B;sKAAA,+B;wKAAA,+B;0KAAA,+B;4KAAA,+B;8KAAA,+B;gLAAA,+B;kLAAA,+B;oLAAA,+B;sLAAA,+B;wLAAA,+B;0LAAA,+B;4LAAA,+B;8LAAA,+B;gMAAA,+B;kMAAA,+B;oMAAA,+B;sMAAA,+B;wMAAA,+B;0MAAA,+B;4MAAA,+B;8MAAA,+B;gNAAA,+B;kNAAA,+B;oNAAA,+B;sNAAA,+B;wNAAA,+B;0NAAA,+B;4NAAA,+B;8NAAA,+B;gOAAA,+B;kOAAA,+B;oOAAA,+B;sOAAA,+B;wOAAA,+B;0OAAA,+B;4OAAA,+B;8OAAA,+B;gPAAA,+B;kPAAA,+B;oPAAA,+B;sPAAA,+B;wPAAA,+B;0PAAA,+B;4PAAA,+B;8PAAA,+B;gQAAA,+B;kQAAA,+B;oQAAA,+B;sQAAA,+B;wQAAA,+B;0QAAA,+B;4QAAA,+B;8QAAA,+B;gRAAA,+B;kRAAA,+B;oRAAA,+B;sRAAA,+B;wRAAA,+B;0RAAA,+B;4RAAA,+B;8RAAA,+B;gSAAA,+B;kSAAA,+B;oSAAA,+B;sSAAA,+B;wSAAA,+B;0SAAA,+B;4SAAA,+B;8SAAA,+B;gTAAA,+B;kTAAA,+B;oTAAA,+B;sTAAA,+B;wTAAA,+B;0TAAA,+B;4TAAA,+B;8TAAA,+B;gUAAA,+B;kUAAA,+B;oUAAA,+B;sUAAA,+B;wUAAA,+B;0UAAA,+B;4UAAA,+B;8UAAA,+B;gVAAA,+B;kVAAA,+B;oVAAA,+B;sVAAA,+B;wVAAA,+B;0VAAA,+B;4VAAA,+B;8VAAA,+B;gWAAA,+B;kWAAA,+B;oWAAA,+B;sWAAA,+B;wWAAA,+B;0WAAA,+B;4WAAA,+B;8WAAA,+B;gXAAA,+B;kXAAA,+B;oXAAA,+B;sXAAA,+B;wXAAA,+B;0XAAA,+B;4XAAA,+B;8XAAA,+B;gYAAA,+B;kYAAA,+B;oYAAA,+B;sYAAA,+B;wYAAA,+B;0YAAA,+B;4YAAA,+B;8YAAA,+B;gZAAA,+B;kZAAA,+B;oZAAA,+B;sZAAA,+B;wZAAA,+B;0ZAAA,+B;4ZAAA,+B;8ZAAA,+B;gaAAA,+B;kaAAA,+B;oaAAA,+B;saAAA,+B;waAAA,+B;0aAAA,+B;4aAAA,+B;8aAAA,+B;gbAAA,+B;kbAAA,+B;obAAA,+B;sbAAA,+B;wbAAA,+B;0bAAA,+B;4bAAA,+B;8bAAA,+B;gcAAA,+B;kcAAA,+B;ocAAA,+B;scAAA,+B;wcAAA,+B;0cAAA,+B;4cAAA,+B;8cAAA,+B;gdAAA,+B;kdAAA,+B;odAAA,+B;sdAAA,+B;wdAAA,+B;0dAAA,+B;4dAAA,+B;8dAAA,+B;geAAA,+B;keAAA,+B;oeAAA,+B;seAAA,+B;weAAA,+B;0eAAA,+B;4eAAA,+B;8eAAA,+B;gfAAA,+B;kfAAA,+B;ofAAA,+B;sfAAA,+B;wfAAA,+B;0fAAA,+B;4fAAA,+B;8fAAA,+B;ggBAAA,+B;kgBAAA,+B;ogBAAA,+B;sgBAAA,+B;wgBAAA,+B;0gBAAA,+B;4gBAAA,+B;8gBAAA,+B;ghBAAA,+B;khBAAA,+B;ohBAAA,+B;shBAAA,+B;whBAAA,+B;0hBAAA,+B;4hBAAA,+B;8hBAAA,+B;giBAAA,+B;kiBAAA,+B;oiBAAA,+B;siBAAA,+B;wiBAAA,+B;0iBAAA,+B;4iBAAA,+B;8iBAAA,+B;gjBAAA,+B;kjBAAA,+B;ojBAAA,+B;sjBAAA,+B;wjBAAA,+B;0jBAAA,+B;4jBAAA,+B;8jBAAA,+B;gkBAAA,+B;kkBAAA,+B;okBAAA,+B;skBAAA,+B;wkBAAA,+B;0kBAAA,+B;4kBAAA,+B;8kBAAA,+B;glBAAA,+B;klBAAA,+B;olBAAA,+B;slBAAA,+B;wlBAAA,+B;0lBAAA,+B;4lBAAA,+B;8lBAAA,+B;gmBAAA,+B;kmBAAA,+B;omBAAA,+B;smBAAA,+B;wmBAAA,+B;0mBAAA,+B;4mBAAA,+B;8mBAAA,+B;gnBAAA,+B;knBAAA,+B;onBAAA,+B;snBAAA,+B;wnBAAA,+B;0nBAAA,+B;4nBAAA,+B;8nBAAA,+B;goBAAA,+B;koBAAA,+B;ooBAAA,+B;soBAAA,+B;woBAAA,+B;0oBAAA,+B;4oBAAA,+B;8oBAAA,+B;gpBAAA,+B;kpBAAA,+B;opBAAA,+B;spBAAA,+B;wpBAAA,+B;0pBAAA,+B;4pBAAA,+B;8pBAAA,+B;gqBAAA,+B;kqBAAA,+B;oqBAAA,+B;sqBAAA,+B;wqBAAA,+B;0qBAAA,+B;4qBAAA,+B;8qBAAA,+B;grBAAA,+B;krBAAA,+B;orBAAA,+B;srBAAA,+B;wrBAAA,+B;0rBAAA,+B;4rBAAA,+B;8rBAAA,+B;gsBAAA,+B;ksBAAA,+B;osBAAA,+B;ssBAAA,+B;wsBAAA,+B;0sBAAA,+B;4sBAAA,+B;8sBAAA,+B;gtBAAA,+B;ktBAAA,+B;otBAAA,+B;stBAAA,+B;wtBAAA,+B;0tBAAA,+B;4tBAAA,+B;8tBAAA,+B;guBAAA,+B;kuBAAA,+B;ouBAAA,+B;suBAAA,+B;wuBAAA,+B;0uBAAA,+B;4uBAAA,+B;8uBAAA,+B;gvBAAA,+B;kvBAAA,+B;ovBAAA,+B;svBAAA,+B;wvBAAA,+B;0vBAAA,+B;4vBAAA,+B;8vBAAA,+B;gwBAAA,+B;kwBAAA,+B;owBAAA,+B;swBAAA,+B;wwBAAA,+B;0wBAAA,+B;4wBAAA,+B;8wBAAA,+B;gxBAAA,+B;kxBAAA,+B;oxBAAA,+B;sxBAAA,+B;wxBAAA,+B;0xBAAA,+B;4xBAAA,+B;8xBAAA,+B;gyBAAA,+B;kyBAAA,+B;oyBAAA,+B;syBAAA,+B;wyBAAA,+B;0yBAAA,+B;4yBAAA,+B;8yBAAA,+B;gzBAAA,+B;kzBAAA,+B;ozBAAA,+B;szBAAA,+B;wzBAAA,+B;0zBAAA,+B;4zBAAA,+B;8zBAAA,+B;g0BAAA,+B;k0BAAA,+B;o0BAAA,+B;s0BAAA,+B;w0BAAA,+B;00BAAA,+B;40BAAA,+B;80BAAA,+B;g1BAAA,+B;k1BAAA,+B;o1BAAA,+B;s1BAAA,+B;w1BAAA,+B;01BAAA,+B;41BAAA,+B;81BAAA,+B;g2BAAA,+B;k2BAAA,+B;o2BAAA,+B;s2BAAA,+B;w2BAAA,+B;02BAAA,+B;42BAAA,+B;82BAAA,+B;g3BAAA,+B;k3BAAA,+B;o3BAAA,+B;s3BAAA,+B;w3BAAA,+B;03BAAA,+B;43BAAA,+B;83BAAA,+B;g4BAAA,+B;k4BAAA,+B;o4BAAA,+B;s4BAAA,+B;w4BAAA,+B;04BAAA,+B;44BAAA,+B;84BAAA,+B;g5BAAA,+B;k5BAAA,+B;o5BAAA,+B;s5BAAA,+B;w5BAAA,+B;05BAAA,+B;45BAAA,+B;85BAAA,+B;g6BAAA,+B;k6BAAA,+B;o6BAAA,+B;s6BAAA,+B;w6BAAA,+B;06BAAA,+B;46BAAA,+B;86BAAA,+B;g7BAAA,+B;k7BAAA,+B;o7BAAA,+B;s7BAAA,+B;w7BAAA,+B;07BAAA,+B;47BAAA,+B;87BAAA,+B;g8BAAA,+B;k8BAAA,+B;o8BAAA,+B;s8BAAA,+B;w8BAAA,+B;08BAAA,+B;48BAAA,+B;88BAAA,+B;g9BAAA,+B;k9BAAA,+B;o9BAAA,+B;s9BAAA,+B;w9BAAA,+B;09BAAA,+B;49BAAA,+B;89BAAA,+B;g+BAAA,+B;k+BAAA,+B;o+BAAA,+B;s+BAAA,+B;w+BAAA,+B;0+BAAA,+B;4+BAAA,+B;8+BAAA,+B;g/BAAA,+B;k/BAAA,+B;o/BAAA,+B;s/BAAA,+B;w/BAAA,+B;0/BAAA,+B;4/BAAA,+B;8/BAAA,+B;gAC

EAAyB,SAAzB,C;MAEA,OAAO,sBAmGmC,CAnG5B,iBAmG6B,GAnGV,KAmGU,IAAD,IAAa,eAnGhD,4D;K;kCAGX,Y;MAAe,qBAAQ,e;K;IAEgB,4D;MAAA,sC;MAAS,2B;MAC5C,eAAoB,oB;MACpB,eAAoB,4B;K;8DAEpB,Y;MAKgB,Q;MAJZ,IAAI,iBAAS,CAAb,C;QACI,W;;QAGA,mBAAQ,sCAAO,YAAP,4DAAR,C;QACA,eAoFkC,CApF1B,YAoF2B,GApFb,CAoFa,IAAD,IAAa,+B;QAnF/C,mC;;K;;oCAXZ,Y;MAAuC,kD;K;2CAgBvC,iB;MAGiE,UAQ1C,MAR0C,EAe1C,MAf0C,EAqBtD,M;MAtBP,aACQ,KAAM,OAAN,GAAa,IAAK,KAAtB,GAAkC,UAAN,KAAM,EAAO,IAAK,KAAZ,CAAlC,GAAyD,kD;MAE7D,WAAW,IAAK,K;MAEhB,WAAW,C;MACX,UAAU,iB;MAEV,OAAO,OAAO,IAAP,IAAe,MAAM,eAA5B,C;QACI,OAAO,IAAP,IAAe,wBAAO,GAAP,gE;QACf,mB;QACA,iB;;MAGJ,MAAM,C;MACN,OAAO,OAAO,IAAd,C;QACI,OAAO,IAAP,IAAe,wBAAO,GAAP,gE;QACf,mB;QACA,iB;;MAEJ,MAAM,IAAI,MAAO,OAAP,GAAgB,GAAK,KAAL,GAAa,IAAK,KAAlB,CAAhB,GAAuC,IAAK,KAAlD

,a;;QACJ,OrK9C+D,aqK8CxD,kBAAM,CAAN,CrK9C0E,KAAlB,EqK8C7C,KrK9C2E,KAA9B,CqK8CxD,IAAP,C;UACI,a;;QACJ,IAAI,KAAK,CAAT,C;UACI,UAAU,kBAAM,CAAN,C;UACV,kBAAM,CAAN,EAAW,kBAAM,CAAN,CAAX,C;UACA,kBAAM,CAAN,EAAW,GAAX,C;UACA,a;UACA,a;;;MAGR,OAAO,C;K;IAGX,yC;MAGI,YAAY,aAAU,KAAV,EAAiB,IAAjB,EAAuB,KAAvB,C;MACZ,IAAI,QAAO,QAAQ,CAAR,IAAP,C;QACI,YAAU,KAAV,EAAiB,IAAjB,EAAuB,QAAQ,CAAR,IAAvB,C;MACJ,IAAI,QAAQ,KAAZ,C;QACI,YAAU,KAAV,EAAiB,KAAjB,EAAwB,KAAxB,C;K;IAKR,gD;MAIE,UAAU,KAAV,EAAiB,SAAjB,EAA4B,EAA4B,UAA5B,CAAuC,KAAvC,C;MAC5B,UAAU,KAAV,EAAiB,SAAjB,EAA4B,EAA4B,UAA5B,CAAuC,KAAvC,C;K;IAGF,2E,gD;MAIE,Y

W,SAAQ,CAAR,EAAW,CAAX,C;O;K;IAZ/B,gC;MAOI,sBAAW,6BAAX,C;K;4FASJ,yB;MAAA,4D;MAAA,wD;MAAA,mB;QAOqE,kBAAW,cAAX,C;O;KAPrE,C;IAgBe,8C;MAAA,uB;QAEH,UAAM,CAAN,C;UADJ,OACe,C;aACX,c;UAFJ,OAEiB,C;aACb,c;UAHJ,OAGiB,E;;UAHjB,OAIY,kBAAW,SAAQ,CAAR,EAAW,CAAX,C;O;K;IAZ/B,+B;MAOI,sBAAW,4BAAX,C;K;0FASJ,yB;MAAA,4D;MAAA,sD;MAAA,mB;QAOoE,iBAAU,cAAV,C;O;KAPpE,C;IASA,wB;MAK4F,Q;MA7B,OAA6B,4F;K;IAE5F,wB;MAK4F,Q;MA7B,OAA6B,4F;K;IAE5F,gC;MAM+D,IAEJ,IAFI,EAGJ,M;MAFvD,kBAD2D,SAC3D,sB;QADqD,OAC5B,sBAAK,W;WAC9B,WAF2D,SAE3D,wC;QAFqD,OAEE,4F;WACvD,WAH2D,SAG3D,wC;QAHqD,OAGE,gG;;QAHF,OAI7C,uBAAAmB,SAAAnB,C;K;IAIuB,wC;MAAC,4B;K;2CAChC,gB;MAAwC,OAAA,eAAW,SAAQ,CAAR,EAAW,CAAX,C;K;4CACnD,Y;MACgC,sB;K;;IAGpC,kC;MAAA,sC;K;+CACI,gB;MAAoE,OAAE,iBAAF,CAAE,EAAU,CAAV,C;K;gDACtE,Y;MAC8C,2C;K;;;IAHlD,8C;MAAA,6C;QAAA,4B;;MAAA,sC;K;IAMA,kC;MAAA,sC;K;+CACI,gB;MAAoE,OAAE,iBAAF,CAAE,EAAU,CAAV,C;K;gDACtE,Y;MAC8C,2C;K;;;IAHlD,8C;MAAA,6C;QAAA,4B;;MAAA,sC;K;8EyKjTA,4B;MAUI,OAAK,iBAAL,CAAK,EAAU,CAAV,C;K;ICTT,iC;K;;;;oDAyDI,0C;MAiB+D,oB;QAAA,4C;2CAAA,gB;MAAA,2C;QAAA,0C;;MAAA,2C;QAAA,0C;;MAAA,2C;QAAA,0C;

mDAC5E,mB;MAAwE,c;K;uDACxE,e;MAA8D,W;K;+CAC9D,Y;MAAsC,Q;K;+CACtC,Y;MAAyC,8B;K;;;IAb7C,6C;MAAA,4C;QAAA,2B;;MAAA,qC;K;IAqB8B,wC;MAC1B,kB;MACA,wB;K;4CAGA,e;MAGQ,Q;MAFJ,UAAU,I;MACV,OAAO,IAAP,C;QACI,YAAA,GAAI,UAAJ,aAAY,GAAZ,W;UAAwB,W;;QACxB,WAAW,GAAI,O;QACf,IAAI,oCAAJ,C;UACI,MAAM,I;;UAEN,OAAO,iBAAK,GAAL,C;;;K;6CAKnB,8B;MACI,iBAAU,WAAK,cAAK,GAAL,cAAK,OAAL,EAAc,SAAd,CAAf,EAAyC,cAAzC,C;K;iDAEJ,e;UAGW,I;MAFP,+BAAQ,GAAR,U;QAAoB,O;UAAO,W;;MAC3B,cAAc,WAAK,kBAAS,GAAT,C;MAEf,gBAAY,WAAZ,C;QAAoB,O;UAAO,W;WACpB,gBAAY,mCAAZ,C;QAAqB,O;UAAO,qB;;QAC5B,2BAAgB,OAAhB,hB,EAAyB,cAAzB,C;MAAgC,WAAhC,K;uCAAoC,OAApC,I,IAAyC,OAAzC,Q;MAAiD,WAAjD,C;QAAoD,O;UAAO,W;WAAP,C;QAAoB,O;UAAO,W;;MAAhF,O;QAAgF,O;UAAO,W;;MAAvF,O;QAAuF,O;UAAO,W;;MAAvG,O;QAAuG,O;UAAO,W;;MAAvH,O;QAAuH,O;UAAO,W;;MAAvI,O;QAAuI,O;UAAO,W;;MAAvJ,O;QAAuJ,O;UAAO,W;;MAAvK,O;QAAuK,O;UAAO,W;;MAAvL,O;QAAuL,O;UAAO,W;;MAAvM,O;QAAuM,O;UAAO,W;;MAAvN,O;QAAuN,O;UAAO,W;;MAAvO,O;QAAuO,O;UAAO,W;;MAAvP,O;QAAuP,O;UAAO,W;;MAAvQ,O;QAAuQ,O;UAAO,W;;MAAvR,O;QAAuR,O;UAAO,W;;MAAvS,O;QAAuS,O;UAAO,W;;MAAvT,O;QAAuT,O;UAAO,W;;MAAvU,O;QAAuU,O;UAAO,W;;MAAvV,O;QAAuV,O;UAAO,W;;MAAvW,O;QAAuW,O;UAAO,W;;MAAvX,O;QAAuX,O;UAAO,W;;MAAvY,O;QAAuY,O;UAAO,W;;MAAvZ,O;QAAuZ,O;UAAO,W;;MAAva,O;QAAua,O;UAAO,W;;MAAvb,O;QAAub,O;UAAO,W;;MAAvc,O;QAAuc,O;UAAO,W;;MAAvd,O;QAAud,O;UAAO,W;;MAAve,O;QAAue,O;UAAO,W;;MAAvf,O;QAAuf,O;UAAO,W;;MAAvgB,O;QAAugB,O;UAAO,W;;MAAvhB,O;QAAuhB,O;UAAO,W;;MAAvhB,O;QAAuhB,O;UAAO,W;;MAAvhB,O;QAAuh

C,C;K;;4EC9BR,wC;MAqBI,OAAO,e;K;4EAGX,+C;MAuBI,cAAI,KAAJ,C;K;4EAIJ,wC;MAmBI,OAAO,cAAI,OAAJ,C;K;4EAGX,+C;MAqBI,cAAI,OAAJ,EAAa,KAAb,C;K;IC/FJ,kB;MA6PI,4B;K;+BAtOA,Y;MAOiC,6BAAS,EAAT,C;K;uCAEjC,iB;MAW2C,4BAAQ,CAAR,EAAW,KAAX,C;K;uCAE3C,uB;MAakB,Q;MAHd,iBAAiB,IAAjB,EAAuB,KAAvB,C;MACA,QAAQ,QAAQ,IAAR,I;MACR,IAAI,IAAI,CAAJ,IAAS,IAAS,MAAK,WAAlB,C;QACc,IAAI,MAAM,CAAC,CAAD,IAAM,OAAY,CAAhB,IAAqB,IAAS,CAAT,IAAc,OAAO,C;;UAEA,K;;YAEI,WAAW,cAAU,KAAK,C;YAC1B,IAAI,OAAO,C;;UACN,gBAAO,CAAP,IAAY,IAAS,CAAT,IAAc,GAAG,CAAhB,IAAqB,CAAC,IAAM,CAAC,IAAM,CAAC,IAAM

GAAa,CAAb,IAArB,C;WAC9B,IAAA,KAAM,MAAN,GAAc,WAAd,C;QAH8C,OAGf,0BAAQ,KAAM,MAAN,GAAc,CAAd,IAAR,EAAyB,KAAM,KAA/B,IAAuC,CAAvC,I;;QAHe,OAAtC,mB;K;IAGZ,oC;MAUI,IAAA,KAAM,UAAN,C;QAAmB,MAAM,gCAAyB,uCAAoC,KAA7D,C;WACzB,IAAA,KAAM,KAAN,+C;QAAiD,OAAlB,lB,2BAAS,KAAM,MAAf,f,EAAsB,KAAM,KAAN,yB,BAAA,a,CAAA,kB,EAAA,C;QAAA,C;;K;IAAA,C;;QAAA,C;;K;IAAA,C;;QAAA,C;;K;IAAA,C;

hC,C;K;yCAIX,Y;MACI,OAAW,cAAJ,GAAe,EAAf,GAAuB,MAAW,SAAN,UAAM,CAAX,QAAqC,SAAb,iBA
Aa,CAArC,I;K;yCAGlC,Y;MAAkC,OAAE,UAAF,qBAAU,iB;K;;IAGhD,kC;MAM6E,2BAAgB,SAAhB,EAAsB,I
AAtB,C;K;;;0DAYzE,iB;MAA2C,qCAAiB,UAAjB,EAAwB,KAAxB,KAAkC,8BAAiB,KAAjB,EAAwB,iBAAxB,
C;K;iDAC7E,Y;MAAkC,QAAC,8BAAiB,UAAjB,EAAwB,iBAAxB,C;K;;IAcR,gD;MAI3B,gBAAqB,K;MACrB,u
BAA4B,Y;K;0FACD,Y;MAAQ,oB;K;iGACD,Y;MAAQ,2B;K;2DAE1C,gB;MAA+D,YAAK,C;K;mDAEpE,iB;M
AAgD,gBAAS,aAAT,IAAmB,SAAS,oB;K;0CAC5E,Y;MAAkC,SAAE,iBAAU,oBAAZ,C;K;yCAElC,iB;MACI,O
AAO,4CAA+B,kBAAa,KAAM,UAAnB,KAC9B,kBAAU,KAAM,SAAhB,IAA0B,yBAAiB,KAAM,gBADnB,CA
A/B,C;K;2CAIX,Y;MACI,OAAW,cAAJ,GAAe,EAAf,GAAuB,MAAY,SAAP,aAAO,CAAZ,QAAuC,SAAd,oBAA
c,CAAvC,I;K;2CAGlC,Y;MAAkC,OAAE,aAAF,qBAAW,oB;K;;IAGjD,oC;MAOqF,6BAAkB,SAAlB,EAAwB,IA
AxB,C;K;IAQvD,+C;MAI1B,gBAAqB,K;MACrB,uBAA4B,Y;K;yFACF,Y;MAAQ,oB;K;gGACD,Y;MAAQ,2B;
K;0DAEzC,gB;MAA6D,YAAK,C;K;kDAElE,iB;MAA+C,gBAAS,aAAT,IAAmB,SAAS,oB;K;yCAC3E,Y;MAAk
C,SAAE,iBAAU,oBAAZ,C;K;wCAElC,iB;MACI,OAAO,2CAA8B,kBAAa,KAAM,UAAnB,KAC7B,kBAAU,KA
AM,SAAhB,IAA0B,yBAAiB,KAAM,gBADpB,CAA9B,C;K;0CAIX,Y;MACI,OAAW,cAAJ,GAAe,EAAf,GAAuB,M
AAY,SAAP,aAAO,CAAZ,QAAuC,SAAd,oBAAc,CAAvC,I;K;0CAGlC,Y;MAAkC,OAAE,aAAF,qBAAW,o
B;K;;IAGjD,oC;MAOkF,4BAAiB,SAAjB,EAAuB,IAAvB,C;K;oFAGlF,8B;MAQI,0BAAmB,2BAAS,OAAT,C;K;
IAGvB,+C;MACI,IAAI,CAAC,UAAL,C;QAAiB,MAAM,gCAAyB,iCAA8B,8B,IAA9B,iBAAzB,C;K;IC5I3B,gC;MA
cW,Q;MADP,IAAI,CAAC,6BAAW,KAAX,CAAL,C;QAAwB,MAAM,uBAAmB,sC/EjBzC,oB+EiByC,CAAnB,C
;;MAC9B,OAAO,sD;K;IAMX,oC;MAakC,Q;MAA9B,OAAW,6BAAW,KAAX,CAAJ,CAAuB,GAAuB,sDAAvB,GAAuC,
I;K;;;;;;;ICvBhB,yC;MA2B9B,uC;MA1BA,wB;MAIA,gB;M7LQA,IAAI,E6LDS,iBAAY,IAAb,MAAuB,iBAAvB,
C7LCR,CAAJ,C;QACI,c6LDQ,iBAAY,IAAhB,GACI,8CADJ,GAGI,sCAA0B,aAA1B,qC;Q7LDR,MAAM,gCAA
yB,OAAQ,WAAjC,C;;K;yC6LKV,Y;MAAwC,Q;MAAA,oB;MACpC,iB;QAD8B,OACtB,G;WACR,oD;QAF8B,O
AEF,SAAL,SAAK,C;WAC5B,6C;QAH8B,OAGd,iBAAK,SAAL,C;WAChB,8C;QAJ8B,OAIb,kBAAM,SAAN,C;;
QAJa,mC;K;IAOlC,qC;MAAA,yC;MACI,IAAG,CAAC,gBAAgB,IAAhB,EAAsB,IAAtB,C;K;iGAQJ,Y;MAAQ,gB;K;4
DAEzC,gB;MAOI,8DAAqC,IAArC,C;K;gEAEJ,gB;MAMI,uDAA8B,IAA9B,C;K;4DAEJ,gB;MAMI,wDAA+B,+B,I
AA/B,C;K;;;IArCR,iDAAqC,gD;MAAA,gB;QAAA,+B;;MAAA,yC;K;;2CAAA,Y;MAWI,oB;K;2CAXJ,Y;MAeI,gB;K;6CA
fJ,0B;MAAA,2BAWI,8CAXJ,EAeI,kCAfJ,C;K;yCAAA,Y;MAAA,c;MAWI,yD;MAIA,qD;MAfJ,a;K;uCAAA,iB;
MAAA,4IAWI,4CAXJ,IAeI,oCAfJ,I;K;ICLA,kC;MAAA,e;MAAA,iB;MAAA,uB;K;IAAA,gC;MAAA,mC;O;MA
YI,4D;MAKA,8C;MAKA,gD;K;;IAVA,2C;MAAA,sB;MAAA,mC;K;;IAKA,oC;MAAA,sB;MAAA,4B;K;;IAKA,
qC;MAAA,sB;MAAA,6B;K;;IAtBJ,4B;MAAA,mG;K;;IAAA,iC;MAAA,a;aAAA,W;UAAA,wC;aAAA,I;UAAA,i
C;aAAA,K;UAAA,kC;;UAAA,6D;;K;;6ECAA,yB;MAAA,4F;MAAA,2B;QASI,MAAM,mCAA8B,0EAA9B,C;O;
KATV,C;ICkCA,+D;MAaW,Q;MAAP,OAAO,8CAAO,KAAP,EAAc,UAAd,EAA0B,QAA1B,oC;QAAiC,MAAM,IAAGX,kC;MAA
IiB,Q;MAAb,wBAAa,KAAb,gB;QAAa,WAAA,KAAb,M;QACI,yBAAO,IAAP,C;;MACJ,OAAO,S;K;mFAAX,qB
;MAGwD,gCAAO,EAAP,C;K;qFAAxD,4B;MAG4E,OAAA,yBAAO,KAAP,EAAc,CALpB,gBAKoB,EAAd,CAAd,C;K;qFAAO
D,4B;MAGmE,OAAA,yBAAO,KAAP,EAAc,CAVX,WAUW,EAAd,CAAd,C;K;IAaxD,wD;MAEQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAFrB,sD;MAIQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAJrB,sD;MAMQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IANrB,sD;MAQQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IARrB,sD;MAUQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAVrB,sD;MAYQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAZrB,sD;MAcQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAdrB,sD;MAgBQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhBrB,sD;MAkBQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlBrB,sD;MAoBQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApBrB,sD;MAsBQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtBrB,sD;MAwBQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxBrB,sD;MA0BQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1BrB,sD;MA4BQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5BrB,sD;MA8BQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9BrB,sD;MAgCQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhCrB,sD;MAkCQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlCrB,sD;MAoCQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApCrB,sD;MAsCQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtCrB,sD;MAwCQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxCrB,sD;MA0CQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1CrB,sD;MA4CQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5CrB,sD;MA8CQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9CrB,sD;MAgDQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhDrB,sD;MAkDQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlDrB,sD;MAoDQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApDrB,sD;MAsDQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtDrB,sD;MAwDQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxDrB,sD;MA0DQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1DrB,sD;MA4DQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5DrB,sD;MA8DQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9DrB,sD;MAgEQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhErB,sD;MAkEQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlErB,sD;MAoEQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApErB,sD;MAsEQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtErB,sD;MAwEQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxErB,sD;MA0EQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1ErB,sD;MA4EQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5ErB,sD;MA8EQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9ErB,sD;MAgFQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhFrB,sD;MAkFQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlFrB,sD;MAoFQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApFrB,sD;MAsFQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtFrB,sD;MAwFQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxFrB,sD;MA0FQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1FrB,sD;MA4FQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5FrB,sD;MA8FQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9FrB,sD;MAgGQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhGrB,sD;MAkGQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlGrB,sD;MAoGQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApGrB,sD;MAsGQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtGrB,sD;MAwGQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxGrB,sD;MA0GQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1GrB,sD;MA4GQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5GrB,sD;MA8GQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9GrB,sD;MAgHQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhHrB,sD;MAkHQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlHrB,sD;MAoHQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApHrB,sD;MAsHQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtHrB,sD;MAwHQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxHrB,sD;MA0HQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1HrB,sD;MA4HQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5HrB,sD;MA8HQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9HrB,sD;MAgIQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhIrB,sD;MAkIQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlIrB,sD;MAoIQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApIrB,sD;MAsIQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtIrB,sD;MAwIQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxIrB,sD;MA0IQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1IrB,sD;MA4IQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5IrB,sD;MA8IQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9IrB,sD;MAgJQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhJrB,sD;MAkJQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlJrB,sD;MAoJQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApJrB,sD;MAsJQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtJrB,sD;MAwJQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxJrB,sD;MA0JQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1JrB,sD;MA4JQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5JrB,sD;MA8JQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9JrB,sD;MAgKQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhKrB,sD;MAkKQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlKrB,sD;MAoKQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApKrB,sD;MAsKQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtKrB,sD;MAwKQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxKrB,sD;MA0KQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1KrB,sD;MA4KQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5KrB,sD;MA8KQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9KrB,sD;MAgLQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhLrB,sD;MAkLQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlLrB,sD;MAoLQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApLrB,sD;MAsLQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtLrB,sD;MAwLQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxLrB,sD;MA0LQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1LrB,sD;MA4LQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5LrB,sD;MA8LQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9LrB,sD;MAgMQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhMrB,sD;MAkMQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlMrB,sD;MAoMQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApMrB,sD;MAsMQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtMrB,sD;MAwMQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxMrB,sD;MA0MQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1MrB,sD;MA4MQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5MrB,sD;MA8MQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9MrB,sD;MAgNQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhNrB,sD;MAkNQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlNrB,sD;MAoNQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApNrB,sD;MAsNQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtNrB,sD;MAwNQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxNrB,sD;MA0NQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1NrB,sD;MA4NQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5NrB,sD;MA8NQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9NrB,sD;MAgOQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhOrB,sD;MAkOQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlOrB,sD;MAoOQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApOrB,sD;MAsOQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtOrB,sD;MAwOQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxOrB,sD;MA0OQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1OrB,sD;MA4OQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5OrB,sD;MA8OQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9OrB,sD;MAgPQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhPrB,sD;MAkPQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlPrB,sD;MAoPQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApPrB,sD;MAsPQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtPrB,sD;MAwPQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxPrB,sD;MA0PQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1PrB,sD;MA4PQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5PrB,sD;MA8PQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9PrB,sD;MAgQQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhQrB,sD;MAkQQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlQrB,sD;MAoQQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApQrB,sD;MAsQQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtQrB,sD;MAwQQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxQrB,sD;MA0QQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1QrB,sD;MA4QQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5QrB,sD;MA8QQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9QrB,sD;MAgRQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhRrB,sD;MAkRQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlRrB,sD;MAoRQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApRrB,sD;MAsRQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtRrB,sD;MAwRQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxRrB,sD;MA0RQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1RrB,sD;MA4RQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5RrB,sD;MA8RQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9RrB,sD;MAgSQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhSrB,sD;MAkSQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlSrB,sD;MAoSQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApSrB,sD;MAsSQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtSrB,sD;MAwSQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxSrB,sD;MA0SQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1SrB,sD;MA4SQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5SrB,sD;MA8SQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9SrB,sD;MAgTQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhTrB,sD;MAkTQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlTrB,sD;MAoTQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApTrB,sD;MAsTQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtTrB,sD;MAwTQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxTrB,sD;MA0TQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1TrB,sD;MA4TQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5TrB,sD;MA8TQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9TrB,sD;MAgUQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhUrB,sD;MAkUQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlUrB,sD;MAoUQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApUrB,sD;MAsUQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtUrB,sD;MAwUQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxUrB,sD;MA0UQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1UrB,sD;MA4UQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5UrB,sD;MA8UQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9UrB,sD;MAgVQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhVrB,sD;MAkVQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlVrB,sD;MAoVQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApVrB,sD;MAsVQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtVrB,sD;MAwVQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxVrB,sD;MA0VQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1VrB,sD;MA4VQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5VrB,sD;MA8VQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9VrB,sD;MAgWQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhWrB,sD;MAkWQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlWrB,sD;MAoWQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApWrB,sD;MAsWQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtWrB,sD;MAwWQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxWrB,sD;MA0WQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1WrB,sD;MA4WQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5WrB,sD;MA8WQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9WrB,sD;MAgXQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhXrB,sD;MAkXQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlXrB,sD;MAoXQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApXrB,sD;MAsXQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtXrB,sD;MAwXQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxXrB,sD;MA0XQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1XrB,sD;MA4XQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5XrB,sD;MA8XQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9XrB,sD;MAgYQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhYrB,sD;MAkYQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlYrB,sD;MAoYQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApYrB,sD;MAsYQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtYrB,sD;MAwYQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxYrB,sD;MA0YQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1YrB,sD;MA4YQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5YrB,sD;MA8YQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9YrB,sD;MAgZQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhZrB,sD;MAkZQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlZrB,sD;MAoZQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApZrB,sD;MAsZQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtZrB,sD;MAwZQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxZrB,sD;MA0ZQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1ZrB,sD;MA4ZQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5ZrB,sD;MA8ZQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9ZrB,sD;MAgaQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAharB,sD;MAkaQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlarB,sD;MAoaQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAparB,sD;MAsaQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtarB,sD;MAwaQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxarB,sD;MA0aQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1arB,sD;MA4aQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5arB,sD;MA8aQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA9arB,sD;MAgbQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAhbrB,sD;MAkbQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAlbrB,sD;MAobQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IApbrB,sD;MAsbQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAtbrB,sD;MAwbQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IAxbrB,sD;MA0bQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA1brB,sD;MA4bQ,sB;QAAqB,qBAAA,KAAA,EAAA,C;;MAAA,OAAA,I;K;IA5brB,sD;MA8bQ,sB;QAAqB,qBAAA,KAAA,EAAA

2B,OFrMyB,oBEqMzB,SFrMyB,CAAY,cAfrB,YAAY,CAAZ,CEoNhB,KFrMyB,oBEqMI,UFrMJ,CAAY,cAfrB,YAAY,CAAZ,C;;MEoNlD,W;K;IAGJ,gC;MAGyC,QAAQ,cAAA,sCAAK,cAAL,EAAoB,sCAAK,cAAzB,CAAR,6B;K;ImL3OzC,6C;MAc6B,4B;QAAA,eAAuB,G;MAChD,wCAAsB,EAAtB,EAA0B,YAA1B,C;K;IAEJ,mE;MAKwC,yB;QAAA,YAAoB,E;MAAI,4B;QAAA,eAAuB,G;MjMGnF,IAAI,CmBwR+C,CAAC,Q8K1R5C,Y9K0R4C,CnBxRpD,C;QACI,ciMHiC,wC;QjMIjC,MAAM,gCAAyB,OAAQ,WAAjC,C,C;;MiMHV,cAAY,gB;MAEC,yBAAS,mBAAS,YAAA,SAAU,OAAV,EAAmB,EAAnB,EAAuB,mBAAS,KAAzB,EAAgC,KAAhC,EAAuC,KAAvC,EAA8C,mBAAS,KAAvD,EAA8D,KAAzE,EAAgF,mBAAS,KAAzF,EAAgG,KAAhG,EAAuG,mBAAS,KAAhH,EAAuH,KAAlI,EAAyI,mBAAS,KAAlJ,EAAyJ,KAAzJ,EAAgK,mBAAS,KAAzK,EAAgL,KAAhL,EAAuL,mBAAS,KAAhM,EAAuM,KAAlN,EAAyN,mBAAS,KAAlO,EAAyO,KAAzO,EAAgP,mBAAS,KAAzP,EAAgQ,KAAhQ,EAAuQ,mBAAS,KAAhR,EAAuR,KAAlS,EAAyS,mBAAS,KAAlT,EAAyT,KAAzT,EAAgU,mBAAS,KAAzU,EAAgV,KAAhV,EAAuV,mBAAS,KAAhW,EAAuW,KAAlX,EAAyX,mBAAS,KAAlY,EAAyY,KAAzY,EAAgZ,mBAAS,KAAzZ,EAAga,KAAha,EAAua,mBAAS,KAAhb,EAAub,KAAlc,EAAyc,mBAAS,KAAld,EAAyd,KAAzd,EAAge,mBAAS,KAAze,EAAgf,KAAhf,EAAuf,mBAAS,KAAhgB,EAAugB,KAAlhB,EAAyhB,mBAAS,KAAliB,EAAyiB,KAAziB,EAAgjB,mBAAS,KAAzjB,EAAgkB,KAAhkB,EAAukB,mBAAS,KAAhlB,EAAulB,KAAlmB,EAAymB,mBAAS,KAAlnB,EAAynB,KAAznB,EAAgoB,mBAAS,KAAzoB,EAAgpB,KAAhpB,EAAupB,mBAAS,KAAhqB,EAAuqB,KAAlrB,EAAyrB,mBAAS,KAAlsB,EAAysB,KAAztB,EAAguB,mBAAS,KAAzuB,EAAgvB,KAAhvB,EAAuvB,mBAAS,KAAhwB,EAAuwB,KAAlxB,EAAyxB,mBAAS,KAAlyB,EAAyyB,KAAzyB,EAAgzB,mBAAS,KAAz0B,EAAg1B,KAAh1B,EAAu1B,mBAAS,KAAh2B,EAAu2B,KAAl3B,EAAy3B,mBAAS,KAAl4B,EAAy4B,KAAz4B,EAAg5B,mBAAS,KAAz5B,EAAg6B,KAAh6B,EAAu6B,mBAAS,KAAh7B,EAAu7B,KAAl8B,EAAy8B,mBAAS,KAAl9B,EAAy9B,KAAz9B,EAAg+B,mBAAS,KAAz+B,EAAg/B,KAAh/B,EAAu/B,mBAAS,KAAhgC,EAAugC,KAAlhC,EAAyhC,mBAAS,KAAliC,EAAyiC,KAAziC,EAAgjC,mBAAS,KAAzjC,EAAgkC,KAAhkC,EAAukC,mBAAS,KAAhlC,EAAulC,KAAlmC,EAAymC,mBAAS,KAAlnC,EAAynC,KAAznC,EAAgoC,mBAAS,KAAzoC,EAAgpC,KAAhpC,EAAupC,mBAAS,KAAhqC,EAAuqC,KAAlrC,EAAyrC,mBAAS,KAAlsC,EAAysC,KAAztC,EAAguC,mBAAS,KAAzuC,EAAgvC,KAAhvC,EAAuvC,mBAAS,KAAhwC,EAAuwC,KAAlxC,EAAyxC,mBAAS,KAAlyC,EAAyyC,KAAzyC,EAAgzC,mBAAS,KAAz0C,EAAg1C,KAAh1C,EAAu1C,mBAAS,KAAh2C,EAAu2C,KAAl3C,EAAy3C,mBAAS,KAAl4C,EAAy4C,KAAz4C,EAAg5C,mBAAS,KAAz5C,EAAg6C,KAAh6C,EAAu6C,mBAAS,KAAh7C,EAAu7C,KAAl8C,EAAy8C,mBAAS,KAAl9C,EAAy9C,KAAz9C,EAAg+C,mBAAS,KAAz+C,EAAg/C,KAAh/C,EAAu/C,mBAAS,KAAhgD,EAAugD,KAAlhD,EAAyhD,mBAAS,KAAliD,EAAyiD,KAAziD,EAAgjD,mBAAS,KAAzjD,EAAgkD,KAAhkD,EAAukD,mBAAS,KAAhlD,EAAulD,KAAlmD,EAAymD,mBAAS,KAAlnD,EAAynD,KAAznD,EAAgoD,mBAAS,KAAzoD,EAAgpD,KAAhpD,EAAupD,mBAAS,KAAhqD,EAAuqD,KAAlrD,EAAyrD,mBAAS,KAAlsD,EAAysD,KAAztD,EAAguD,mBAAS,KAAzuD,EAAgvD,KAAhvD,EAAuvD,mBAAS,KAAhwD,EAAuwD,KAAlxD,EAAyxD,mBAAS,KAAlyD,EAAyyD,KAAzyD,EAAgzD,mBAAS,KAAz0D,EAAg1D,KAAh1D,EAAu1D,mBAAS,KAAh2D,EAAu2D,KAAl3D,EAAy3D,mBAAS,KAAl4D,EAAy4D,KAAz4D,EAAg5D,mBAAS,KAAz5D,EAAg6D,KAAh6D,EAAu6D,mBAAS,KAAh7D,EAAu7D,KAAl8D,EAAy8D,mBAAS,KAAl9D,EAAy9D,KAAz9D,EAAg+D,mBAAS,KAAz+D,EAAg/D,KAAh/D,EAAu/D,mBAAS,KAAhgE,EAAugE,KAAlhE,EAAyhE,mBAAS,KAAliE,EAAyiE,KAAziE,EAAgjE,mBAAS,KAAzjE,EAAgkE,KAAhkE,EAAukE,mBAAS,KAAhlE,EAAulE,KAAlmE,EAAymE,mBAAS,KAAlnE,EAAynE,KAAznE,EAAgoE,mBAAS,KAAzoE,EAAgpE,KAAhpE,EAAupE,mBAAS,KAAhqE,EAAuqE,KAAlrE,EAAyrE,mBAAS,KAAlsE,EAAysE,KAAztE,EAAguE,mBAAS,KAAzuE,EAAgvE,KAAhvE,EAAuvE,mBAAS,KAAhwE,EAAuwE,KAAlxE,EAAyxE,mBAAS,KAAlyE,EAAyyE,KAAzyE,EAAgzE,mBAAS,KAAz0E,EAAg1E,KAAh1E,EAAu1E,mBAAS,KAAh2E,EAAu2E,KAAl3E,EAAy3E,mBAAS,KAAl4E,EAAy4E,KAAz4E,EAAg5E,mBAAS,KAAz5E,EAAg6E,KAAh6E,EAAu6E,mBAAS,KAAh7E,EAAu7E,KAAl8E,EAAy8E,mBAAS,KAAl9E,EAAy9E,KAAz9E,EAAg+E,mBAAS,KAAz+E,EAAg/E,KAAh/E,EAAu/E,mBAAS,KAAhgF,EAAugF,KAAlhF,EAAyhF,mBAAS,KAAliF,EAAyiF,KAAziF,EAAgjF,mBAAS,KAAzjF,EAAgkF,KAAhkF,EAAukF,mBAAS,KAAhlF,EAAulF,KAAlmF,EAAymF,mBAAS,KAAlnF,EAAynF,KAAznF,EAAgoF,mBAAS,KAAzoF,EAAgpF,KAAhpF,EAAupF,mBAAS,KAAhqF,EAAuqF,KAAlrF,EAAyrF,mBAAS,KAAlsF,EAAysF,KAAztF,EAAguF,mBAAS,KAAzuF,EAAgvF,KAAhvF,EAAuvF,mBAAS,KAAhwF,EAAuwF,KAAlxF,EAAyxF,mBAAS,KAAlyF,EAAyyF,KAAzyF,EAAgzF,mBAAS,KAAz0F,EAAg1F,KAAh1F,EAAu1F,mBAAS,KAAh2F,EAAu2F,KAAl3F,EAAy3F,mBAAS,KAAl4F,EAAy4F,KAAz4F,EAAg5F,mBAAS,KAAz5F,EAAg6F,KAAh6F,EAAu6F,mBAAS,KAAh7F,EAAu7F,KAAl8F,EAAy8F,mBAAS,KAAl9F,EAAy9F,KAAz9F,EAAg+F,mBAAS,KAAz+F,EAAg/F,KAAh/F,EAAu/F,mBAAS,KAAhgG,EAAugG,KAAlhG,EAAyhG,mBAAS,KAAliG,EAAyiG,KAAziG,EAAgjG,mBAAS,KAAzjG,EAAgkG,KAAhkG,EAAukG,mBAAS,KAAhlG,EAAulG,KAAlmG,EAAymG,mBAAS,KAAlnG,EAAynG,KAAznG,EAAgoG,mBAAS,KAAzoG,EAAgpG,KAAhpG,EAAupG,mBAAS,KAAhqG,EAAuqG,KAAlrG,EAAyrG,mBAAS,KAAlsG,EAAysG,KAAztG,EAAguG,mBAAS,KAAzuG,EAAgvG,KAAhvG,EAAuvG,mBAAS,KAAhwG,EAAuwG,KAAlxG,EAAyxG,mBAAS,KAAlyG,EAAyyG,KAAzyG,EAAgzG,mBAAS,KAAz0G,EAAg1G,KAAh1G,EAAu1G,mBAAS,KAAh2G,EAAu2G,KAAl3G,EAAy3G,mBAAS,KAAl4G,EAAy4G,KAAz4G,EAAg5G,mBAAS,KAAz5G,EAAg6G,KAAh6G,EAAu6G,mBAAS,KAAh7G,EAAu7G,KAAl8G,EAAy8G,mBAAS,KAAl9G,EAAy9G,KAAz9G,EAAg+G,mBAAS,KAAz+G,EAAg/G,KAAh/G,EAAu/G,mBAAS,KAAhgH,EAAugH,KAAlhH,EAAyhH,mBAAS,KAAliH,EAAyiH,KAAziH,EAAgjH,mBAAS,KAAzjH,EAAgkH,KAAhkH,EAAukH,mBAAS,KAAhlH,EAAulH,KAAlmH,EAAymH,mBAAS,KAAlnH,EAAynH,KAAznH,EAAgoH,mBAAS,KAAzoH,EAAgpH,KAAhpH,EAAupH,mBAAS,KAAhqH,EAAuqH,KAAlrH,EAAyrH,mBAAS,KAAlsH,EAAysH,KAAztH,EAAguH,mBAAS,KAAzuH,EAAgvH,KAAhvH,EAAuvH,mBAAS,KAAhwH,EAAuwH,KAAlxH,EAAyxH,mBAAS,KAAlyH,EAAyyH,KAAzyH,EAAgzH,mBAAS,KAAz0H,EAAg1H,KAAh1H,EAAu1H,mBAAS,KAAh2H,EAAu2H,KAAl3H,EAAy3H,mBAAS,KAAl4H,EAAy4H,KAAz4H,EAAg5H,mBAAS,KAAz5H,EAAg6H,KAAh6H,EAAu6H,mBAAS,KAAh7H,EAAu7H,KAAl8H,EAAy8H,mBAAS,KAAl9H,EAAy9H,KAAz9H,EAAg+H,mBAAS,KAAz+H,EAAg/H,KAAh/H,EAAu/H,mBAAS,KAAhgI,EAAugI,KAAlhI,EAAyhI,mBAAS,KAAliI,EAAyiI,KAAziI,EAAgjI,mBAAS,KAAzjI,EAAgkI,KAAhkI,EAAukI,mBAAS,KAAhlI,EAAulI,KAAlmI,EAAymI,mBAAS,KAAlnI,EAAynI,KAAznI,EAAgoI,mBAAS,KAAzoI,EAAgpI,KAAhpI,EAAupI,mBAAS,KAAhqI,EAAuqI,KAAlrI,EAAyrI,mBAAS,KAAlsI,EAAysI,KAAztI,EAAguI,mBAAS,KAAzuI,EAAgvI,KAAhvI,EAAuvI,mBAAS,KAAhwI,EAAuwI,KAAlxI,EAAyxI,mBAAS,KAAlyI,EAAyyI,KAAzyI,EAAgzI,mBAAS,KAAz0I,EAAg1I,KAAh1I,EAAu1I,mBAAS,KAAh2I,EAAu2I,KAAl3I,EAAy3I,mBAAS,KAAl4I,EAAy4I,KAAz4I,EAAg5I,mBAAS,KAAz5I,EAAg6I,KAAh6I,EAAu6I,mBAAS,KAAh7I,EAAu7I,KAAl8I,EAAy8I,mBAAS,KAAl9I,EAAy9I,KAAz9I,EAAg+I,mBAAS,KAAz+I,EAAg/I,KAAh/I,EAAu/I,mBAAS,KAAhgJ,EAAugJ,KAAlhJ,EAAyhJ,mBAAS,KAAliJ,EAAyiJ,KAAziJ,EAAgjJ,mBAAS,KAAzjJ,EAAgkJ,KAAhkJ,EAAukJ,mBAAS,KAAhlJ,EAAulJ,KAAlmJ,EAAymJ,mBAAS,KAAlnJ,EAAynJ,KAAznJ,EAAgoJ,mBAAS,KAAzoJ,EAAgpJ,KAAhpJ,EAAupJ,mBAAS,KAAhqJ,EAAuqJ,KAAlrJ,EAAyrJ,mBAAS,KAAlsJ,EAAysJ,KAAztJ,EAAguJ,mBAAS,KAAzuJ,EAAgvJ,KAAhvJ,EAAuvJ,mBAAS,KAAhwJ,EAAuwJ,KAAlxJ,EAAyxJ,mBAAS,KAAlyJ,EAAyyJ,KAAzyJ,EAAgzJ,mBAAS,KAAz0J,EAAg1J,KAAh1J,EAAu1J,mBAAS,KAAh2J,EAAu2J,KAAl3J,EAAy3J,mBAAS,KAAl4J,EAAy4J,KAAz4J,EAAg5J,mBAAS,KAAz5J,EAAg6J,KAAh6J,EAAu6J,mBAAS,KAAh7J,EAAu7J,KAAl8J,EAAy8J,mBAAS,KAAl9J,EAAy9J,KAAz9J,EAAg+J,mBAAS,KAAz+J,EAAg/J,KAAh/J,EAAu/J,mBAAS,KAAhgK,EAAugK,KAAlhK,EAAyhK,mBAAS,KAAliK,EAAyiK,KAAziK,EAAgjK,mBAAS,KAAzjK,EAAgkK,KAAhkK,EAAukK,mBAAS,KAAhlK,EAAulK,KAAlmK,EAAymK,mBAAS,KAAlnK,EAAynK,KAAznK,EAAgoK,mBAAS,KAAzoK,EAAgpK,KAAhpK,EAAupK,mBAAS,KAAhqK,EAAuqK,KAAlrK,EAAyrK,mBAAS,KAAlsK,EAAysK,KAAztK,EAAguK,mBAAS,KAAzuK,EAAgvK,KAAhvK,EAAuvK,mBAAS,KAAhwK,EAAuwK,KAAlxK,EAAyxK,mBAAS,KAAlyK,EAAyyK,KAAzyK,EAAgzK,mBAAS,KAAz0K,EAAg1K,KAAh1K,EAAu1K,mBAAS,KAAh2K,EAAu2K,KAAl3K,EAAy3K,mBAAS,KAAl4K,EAAy4K,KAAz4K,EAAg5K,mBAAS,KAAz5K,EAAg6K,KAAh6K,EAAu6K,mBAAS,KAAh7K,EAAu7K,KAAl8K,EAAy8K,mBAAS,KAAl9K,EAAy9K,KAAz9K,EAAg+K,mBAAS,KAAz+K,EAAg/K,KAAh/K,EAAu/K,mBAAS,KAAhgL,EAAugL,KAAlhL,EAAyhL,mBAAS,KAAliL,EAAyiL,KAAziL,EAAgjL,mBAAS,KAAzjL,EAAgkL,KAAhkL,EAAukL,mBAAS,KAAhlL,EAAulL,KAAlmL,EAAymL,mBAAS,KAAlnL,EAAynL,KAAznL,EAAgoL,mBAAS,KAAzoL,EAAgpL,KAAhpL,EAAupL,mBAAS,KAAhqL,EAAuqL,KAAlrL,EAAyrL,mBAAS,KAAlsL,EAAysL,KAAztL,EAAguL,mBAAS,KAAzuL,EAAgvL,KAAhvL,EAAuvL,mBAAS,KAAhwL,EAAuwL,KAAlxL,EAAyxL,mBAAS,KAAlyL,EAAyyL,KAAzyL,EAAgzL,mBAAS,KAAz0L,EAAg1L,KAAh1L,EAAu1L,mBAAS,KAAh2L,EAAu2L,KAAl3L,EAAy3L,mBAAS,KAAl4L,EAAy4L,KAAz4L,EAAg5L,mBAAS,KAAz5L,EAAg6L,KAAh6L,EAAu6L,mBAAS,KAAh7L,EAAu7L,KAAl8L,EAAy8L,mBAAS,KAAl9L,EAAy9L,KAAz9L,EAAg+L,mBAAS,KAAz+L,EAAg/L,KAAh/L,EAAu/L,mBAAS,KAAhgM,EAAugM,KAAlhM,EAAyhM,mBAAS,KAAliM,EAAyiM,KAAziM,EAAgjM,mBAAS,KAAzjM,EAAgkM,KAAhkM,EAAukM,mBAAS,KAAhlM,EAAulM,KAAlmM,EAAymM,mBAAS,KAAlnM,EAAynM,KAAznM,EAAgoM,mBAAS,KAAzoM,EAAgpM,KAAhpM,EAAupM,mBAAS,KAAhqM,EAAuqM,KAAlrM,EAAyrM,mBAAS,KAAlsM,EAAysM,KAAztM,EAAguM,mBAAS,KAAzuM,EAAgvM,KAAhvM,EAAuvM,mBAAS,KAAhwM,EAAuwM,KAAlxM,EAAyxM,mBAAS,KAAlyM,EAAyyM,KAAzyM,EAAgzM,mBAAS,KAAz0M,EAAg1M,KAAh1M,EAAu1M,mBAAS,KAAh2M,EAAu2M,KAAl3M,EAAy3M,mBAAS,KAAl4M,EAAy4M,KAAz4M,EAAg5M,mBAAS,KAAz5M,EAAg6M,KAAh6M,EAAu6M,mBAAS,KAAh7M,EAAu7M,KAAl8M,EAAy8M,mBAAS,KAAl9M,EAAy9M,KAAz9M,EAAg+M,mBAAS,KAAz+M,EAAg/M,KAAh/M,EAAu/M,mBAAS,KAAhgN,EAAugN,KAAlhN,EAAyhN,mBAAS,KAAliN,EAAyiN,KAAziN,EAAgjN,mBAAS,KAAzjN,EAAgkN,KAAhkN,EAAukN,mBAAS,KAAhlN,EAAulN,KAAlmN,EAAymN,mBAAS,KAAlnN,EAAynN,KAAznN,EAAgoN,mBAAS,KAAzoN,EAAgpN,KAAhpN,EAAupN,mBAAS,KAAhqN,EAAuqN,KAAlrN,EAAyrN,mBAAS,KAAlsN,EAAysN,KAAztN,EAAguN,mBAAS,KAAzuN,EAAgvN,KAAhvN,EAAuvN,mBAAS,KAAhwN,EAAuwN,KAAlxN,EAAyxN,mBAAS,KAAlyN,EAAyyN,KAAzyN,EAAgzN,mBAAS,KAAz0N,EAAg1N,KAAh1N,EAAu1N,mBAAS,KAAh2N,EAAu2N,KAAl3N,EAAy3N,mBAAS,KAAl4N,EAAy4N,KAAz4N,EAAg5N,mBAAS,KAAz5N,EAAg6N,KAAh6N,EAAu6N,mBAAS,KAAh7N,EAAu7N,KAAl8N,EAAy8N,mBAAS,KAAl9N,EAAy9N,KAAz9N,EAAg+N,mBAAS,KAAz+N,EAAg/N,KAAh/N,EAAu/N,mBAAS,KAAhgO,EAAugO,KAAlhO,EAAyhO,mBAAS,KAAliO,EAAyiO,KAAziO,EAAgjO,mBAAS,KAAzjO,EAAgkO,KAAhkO,EAAukO,mBAAS,KAAhlO,EAAulO,KAAlmO,EAAymO,mBAAS,KAAlnO,EAAynO,KAAznO,EAAgoO,mBAAS,KAAzoO,EAAgpO,KAAhpO,EAAupO,mBAAS,KAAhqO,EAAuqO,KAAlrO,EAAyrO,mBAAS,KAAlsO,EAAysO,KAAztO,EAAguO,mBAAS,KAAzuO,EAAgvO,KAAhvO,EAAuvO,mBAAS,KAAhwO,EAAuwO,KAAlxO,EAAyxO,mBAAS,KAAlyO,EAAyyO,KAAzyO,EAAgzO,mBAAS,KAAz0O,EAAg1O,KAAh1O,EAAu1O,mBAAS,KAAh2O,EAAu2O,KAAl3O,EAAy3O,mBAAS,KAAl4O,EAAy4O,KAAz4O,EAAg5O,mBAAS,KAAz5O,EAAg6O,KAAh6O,EAAu6O,mBAAS,KAAh7O,EAAu7O,KAAl8O,EAAy8O,mBAAS,KAAl9O,EAAy9O,KAAz9O,EAAg+O,mBAAS,KAAz+O,EAAg/O,KAAh/O,EAAu/O,mBAAS,KAAhgP,EAAugP,KAAlhP,EAAyhP,mBAAS,KAAliP,EAAyiP,KAAziP,EAAgjP,mBAAS,KAAzjP,EAAgkP,KAAhkP,EAAukP,mBAAS,KAAhlP,EAAulP,KAAlmP,EAAymP,mBAAS,KAAlnP,EAAynP,KAAznP,EAAgoP,mBAAS,KAAzoP,EAAgpP,KAAhpP,EAAupP,mBAAS,KAAhqP,EAAuqP,KAAlrP,EAAyrP,mBAAS,KAAlsP,EAAysP,KAAztP,EAAguP,mBAAS,KAAzuP,EAAgvP,KAAhvP,EAAuvP,mBAAS,KAAhwP,EAAuwP,KAAlxP,EAAyxP,mBAAS,KAAlyP,EAAyyP,KAAzyP,EAAgzP,mBAAS,KAAz0P,EAAg1P,KAAh1P,EAAu1P,mBAAS,KAAh2P,EAAu2P,KAAl3P,EAAy3P,mBAAS,KAAl4P,EAAy4P,KAAz4P,EAAg5P,mBAAS,KAAz5P,EAAg6P,KAAh6P,EAAu6P,mBAAS,KAAh7P,EAAu7P,KAAl8P,EAAy8P,mBAAS,KAAl9P,EAAy9P,KAAz9P,EAAg+P,mBAAS,KAAz+P,EAAg/P,KAAh/P,EAAu/P,mBAAS,KAAhgQ,EAAugQ,KAAlhQ,EAAyhQ,mBAAS,KAAliQ,EAAyiQ,KAAziQ,EAAgjQ,mBAAS,KAAzjQ,EAAgkQ,KAAhkQ,EAAukQ,mBAAS,KAAhlQ,EAAulQ,KAAlmQ,EAAymQ,mBAAS,KAAlnQ,EAAynQ,KAAznQ,EAAgoQ,mBAAS,KAAzoQ,EAAgpQ,KAAhpQ,EAAupQ,mBAAS,KAAhqQ,EAAuqQ,KAAlrQ,EAAyrQ,mBAAS,KAAlsQ,EAAysQ,KAAztQ,EAAguQ,mBAAS,KAAzuQ,EAAgvQ,KAAhvQ,EAAuvQ,mBAAS,KAAhwQ,EAAuwQ,KAAlxQ,EAAyxQ,mBAAS,KAAlyQ,EAAyyQ,KAAzyQ,EAAgzQ,mBAAS,KAAz0Q,EAAg1Q,KAAh1Q,EAAu1Q,mBAAS,KAAh2Q,EAAu2Q,KAAl3Q,EAAy3Q,mBAAS,KAAl4Q,EAAy4Q,KAAz4Q,EAAg5Q,mBAAS,KAAz5Q,EAAg6Q,KAAh6Q,EAAu6Q,mBAAS,KAAh7Q,EAAu7Q,KAAl8Q,EAAy8Q,mBAAS,KAAl9Q,EAAy9Q,KAAz9Q,EAAg+Q,mBAAS,KAAz+Q,EAAg/Q,KAAh/Q,EAAu/Q,mBAAS,KAAhgR,EAAugR,KAAlhR,EAAyhR,mBAAS,KAAliR,EAAyiR,KAAziR,EAAgjR,mBAAS,KAAzjR,EAAgkR,KAAhkR,EAAukR,mBAAS,KAAhlR,EAAulR,KAAlmR,EAAymR,mBAAS,KAAlnR,EAAynR,KAAznR,EAAgoR,mBAAS,KAAzoR,EAAgpR,KAAhpR,EAAupR,mBAAS,KAAhqR,EAAuqR,KAAlrR,EAAyrR,mBAAS,KAAlsR,EAAysR,KAAztR,EAAguR,mBAAS,KAAzuR,EAAgvR,KAAhvR,EAAuvR,mBAAS,KAAhwR,EAAuwR,KAAlxR,EAAyxR,mBAAS,KAAlyR,EAAyyR,KAAzyR,EAAgzR,mBAAS,KAAz0R,EAAg1R,KAAh1R,EAAu1R,mBAAS,KAAh2R,EAAu2R,KAAl3R,EAAy3R,mBAAS,KAAl4R,EAAy4R,KAAz4R,EAAg5R,mBAAS,KAAz5R,EAAg6R,KAAh6R,EAAu6R,mBAAS,KAAh7R,EAAu7R,KAAl8R,EAAy8R,mBAAS,KAAl9R,EAAy9R,KAAz9R,EAAg+R,mBAAS,KAAz+R,EAAg/R,KAAh/R,EAAu/R,mBAAS,KAAhgS,EAAugS,KAAlhS,EAAyhS,mBAAS,KAAliS,EAAyiS,KAAziS,EAAgjS,mBAAS,KAAzjS,EAAgkS,KAAhkS,EAAukS,mBAAS,KAAhlS,EAAulS,KAAlmS,EAAymS,mBAAS,KAAlnS,EAAynS,KAAznS,EAAgoS,mBAAS,KAAzoS,EAAgpS,KAAhpS,EAAupS,mBAAS,KAAhqS,EAAuqS,KAAlrS,EAAyrS,mBAAS,KAAlsS,EAAysS,KAAztS,EAAguS,mBAAS,KAAzuS,EAAgvS,KAAhvS,EAAuvS,mBAAS,KAAhwS,EAAuwS,KAAlxS,EAAyxS,mBAAS,KAAlyS,EAAyyS,KAAzyS,EAAgzS,mBAAS,KAAz0S,EAAg1S,KAAh1S,EAAu1S,mBAAS,KAAh2S,EAAu2S,KAAl3S,EAAy3S,mBAAS,KAAl4S,EAAy4S,KAAz4S,EAAg5S,mBAAS,KAAz5S,EAAg6S,KAAh6S,EAAu6S,mBAAS,KAAh7S,EAAu7S,KAAl8S,EAAy8S,mBAAS,KAAl9S,EAAy9S,KAAz9S,EAAg+S,mBAAS,KAAz+S,EAAg/S,KAAh/S,EAAu/S,mBAAS,KAAhgT,EAAugT,KAAlhT,EAAyhT,mBAAS,KAAliT,EAAyiT,KAAziT,EAAgjT,mBAAS,KAAzjT,EAAgkT,KAAhkT,EAAukT,mBAAS,KAAhlT,EAAulT,KAAlmT,EAAymT,mBAAS,KAAlnT,EAAynT,KAAznT,EAAgoT,mBAAS,KAAzoT,EAAgpT,KAAhpT,EAAupT,mBAAS,KAAhqT,EAAuqT,KAAlrT,EAAyrT,mBAAS,KAAlsT,EAAysT,KAAztT,EAAguT,mBAAS,KAAzuT,EAAgvT,KAAhvT,EAAuvT,mBAAS,KAAhwT,EAAuwT,KAAlxT,EAAyxT,mBAAS,KAAlyT,EAAyyT,KAAzyT,EAAgzT,mBAAS,KAAz0T,EAAg1T,KAAh1T,EAAu1T,mBAAS,KAAh2T,EAAu2T,KAAl3T,EAAy3T,mBAAS,KAAl4T,EAAy4T,KAAz4T,EAAg5T,mBAAS,KAAz5T,EAAg6T,KAAh6T,EAAu6T,mBAAS,KAAh7T,EAAu7T,KAAl8T,EAAy8T,mBAAS,KAAl9T,EAAy9T,KAAz9T,EAAg+T,mBAAS,KAAz+T,EAAg/T,KAAh/T,EAAu/T,mBAAS,KAAhgU,EAAugU,KAAlhU,EAAyhU,mBAAS,KAAliU,EAAyiU,KAAziU,EAAgjU,mBAAS,KAAzjU,EAAgkU,KAAhkU,EAAukU,mBAAS,KAAhlU,EAAulU,KAAlmU,EAAymU,mBAAS,KAAlnU,EAAynU,KAAznU,EAAgoU,mBAAS,KAAzoU,EAAgpU,KAAhpU,EAAupU,mBAAS,KAAhqU,EAAuqU,KAAlrU,EAAyrU,mBAAS,KAAlsU,EAAysU,KAAztU,EAAguU,mBAAS,KAAzuU,EAAgvU,KAAhvU,EAAuvU,mBAAS,KAAhwU,EAAuwU,KAAlxU,EAAyxU,mBAAS,KAAlyU,EAAyyU,KAAzyU,EAAgzU,mBAAS,KAAz0U,EAAg1U,KAAh1U,EAAu1U,mBAAS,KAAh2U,EAAu2U,KAAl3U,EAAy3U,mBAAS,KAAl4U,EAAy4U,KAAz4U,EAAg5U,mBAAS,KAAz5U,EAAg6U,KAAh6U,EAAu6U,mBAAS,KAAh7U,EAAu7U,KAAl8U,EAAy8U,mBAAS,KAAl9U,EAAy9U,KAAz9U,EAAg+U,mBAAS,KAAz+U,EAAg/U,KAAh/U,EAAu/U,mBAAS,KAAhgV,EAAugV,KAAlhV,EAAyhV,mBAAS,KAAliV,EAAyiV,KAAziV,EAAgjV,mBAAS,KAAzjV,EAAgkV,KAAhkV,EAAukV,mBAAS,KAAhlV,EAAulV,KAAlmV,EAAymV,mBAAS,KAAlnV,EAAynV,KAAznV,EAAgoV,mBAAS,KAAzoV,EAAgpV,KAAhpV,EAAupV,mBAAS,KAAhqV,EAAuqV,KAAlrV,EAAyrV,mBAAS,KAAlsV,EAAysV,KAAztV,EAAguV,mBAAS,KAAzuV,EAAgvV,KAAhvV,EAAuvV,mBAAS,KAAhwV,EAAuwV,KAAlxV,EAAyxV,mBAAS,KAAlyV,EAAyyV,KAAzyV,EAAgzV,mBAAS,KAAz0V,EAAg1V,KAAh1V,EAAu1V,mBAAS,KAAh2V,EAAu2V,KAAl3V,EAAy3V,mBAAS,KAAl4V,EAAy4V,KAAz4V,EAAg5V,mBAAS,KAAz5V,EAAg6V,KAAh6V,EAAu6V,mBAAS,KAAh7V,EAAu7V,KAAl8V,EAAy8V,mBAAS,KAAl9V,EAAy9V,KAAz9V,EAAg+V,mBAAS,KAAz+V,EAAg/V,KAAh/V,EAAu/V,mBAAS,KAAhgW,EAAugW,KAAlhW,EAAyhW,mBAAS,KAAliW,EAAyiW,KAAziW,EAAgjW,mBAAS,KAAzjW,EAAgkW,KAAhkW,EAAukW,mBAAS,KAAhlW,EAAulW,KAAlmW,EAAymW,mBAAS,KAAlnW,EAAynW,KAAznW,EAAgoW,mBAAS,KAAzoW,EAAgpW,KAAhpW,EAAupW,mBAAS,KAAhqW,EAAuqW,KAAlrW,EAAyrW,mBAAS,KAAlsW,EAAysW,KAAztW,EAAguW,mBAAS,KAAzuW,EAAgvW,KAAhvW,EAAuvW,mBAAS,KAAhwW,EAAuwW,KAAlxW,EAAyxW,mBAAS,KAAlyW,EAAyyW,KAAzyW,EAAgzW,mBAAS,KAAz0W,EAAg1W,KAAh1W,EAAu1W,mBAAS,KAAh2W,EAAu2W,KAAl3W,EAAy3W,mBAAS,KAAl4W,EAAy4W,KAAz4W,EAAg5W,mBAAS,KAAz5W,EAAg6W,KAAh6W,EAAu6W,mBAAS,KAAh7W,EAAu7W,KAAl8W,EAAy8W,mBAAS,KAAl9W,EAAy9W,KAAz9W,EAAg+W,mBAAS,KAAz+W,EAAg/W,KAAh/W,EAAu/W,mBAAS,KAAhgX,EAAugX,KAAlhX,EAAyhX,mBAAS,KAAliX,EAAyiX,KAAziX,EAAgjX,mBAAS,KAAzjX,EAAgkX,KAAhkX,EAAukX,mBAAS,KAAhlX,EAAulX,KAAlmX,EAAymX,mBAAS,KAAlnX,EAAynX,KAAznX,EAAgoX,mBAAS,KAAzoX,EAAgpX,KAAhpX,EAAupX,mBAAS,KAAhqX,EAAuqX,KAAlrX,EAAyrX,mBAAS,KAAlsX,EAAysX,KAAztX,EAAguX,mBAAS,KAAzuX,EAAgvX,KAAhvX,EAAuvX,mBAAS,KAAhwX,EAAuwX,KAAlxX,EAAyxX,mBAAS,KAAlyX,EAAyyX,KAAzyX,EAAgzX,mBAAS,KAAz0X,EAAg1X,KAAh1X,EAAu1X,mBAAS,KAAh2X,EAAu2X,KAAl3X,EAAy3X,mBAAS,KAAl4X,EAAy4X,KAAz4X,EAAg5X,mBAAS,KAAz5X,EAAg6X,KAAh6X,EAAu6X,mBAAS,KAAh7X,EAAu7X,KAAl8X,EAAy8X,mBAAS,KAAl9X,EAAy9X,KAAz9X,EAAg+X,mBAAS,KAAz+X,EAAg/X,KAAh/X,EAAu/X,mBAAS,KAAhgY,EAAugY,KAAlhY,EAAyhY,mBAAS,KAAliY,EAAyiY,KAAziY,EAAgjY,mBAAS,KAAzjY,EAAgkY,KAAhkY,EAAukY,mBAAS,KAAhlY,EAAulY,KAAlmY,EAAymY,mBAAS,KAAlnY,EAAynY,KAAznY,EAAgoY,mBAAS,KAAzoY,EAAgpY,KAAhpY,EAAupY,mBAAS,KAAhqY,EAAuqY,KAAlrY,EAAyrY,mBAAS,KAAlsY,EAAysY,KAAztY,EAAguY,mBAAS,KAAzuY,EAAgvY,KAAhvY,EAAuvY,mBAAS,KAAhwY,EAAuwY,KAAlxY,EAAyxY,mBAAS,KAAlyY,EAAyyY,KAAzyY,EAAgzY,mBAAS,KAAz0Y,EAAg1Y,KAAh1Y,EAAu1Y,mBAAS,KAAh2Y,EAAu2Y,KAAl3Y,EAAy3Y,mBAAS,KAAl4Y,EAAy4Y,KAAz4Y,EAAg5Y,mBAAS,KAAz5Y,EAAg6Y,KAAh6Y,EAAu6Y,mBAAS,KAAh7Y,EAAu7Y,KAAl8Y,EAAy8Y,mBAAS,KAAl9Y,EAAy9Y,KAAz9Y,EAAg+Y,mBAAS,KAAz+Y,EAAg/Y,KAAh/Y,EAAu/Y,mBAAS,KAAhgZ,EAAugZ,KAAlhZ,EAAyhZ,mBAAS,KAAliZ,EAAyiZ,KAAziZ,EAAgjZ,mBAAS,KAAzjZ,EAAgkZ,KAAhkZ,EAAukZ,mBAAS,KAAhlZ,EAAulZ,KAAlmZ,EAAymZ,mBAAS,KAAlnZ,EAAynZ,KAAznZ,EAAgoZ,mBAAS,KAAzoZ,EAAgpZ,KAAhpZ,EAAupZ,mBAAS,KAAhqZ,EAAuqZ,KAAlrZ,EAAyrZ,mBAAS,KAAlsZ,EAAysZ,KAAztZ,EAAguZ,mBAAS,KAAzuZ,EAAgvZ,KAAhvZ,EAAuvZ,mBAAS,KAAhwZ,EAAuwZ,KAAlxZ,EAAyxZ,mBAAS,KAAlyZ,EAAyyZ,KAAzyZ,EAAgzZ,mBAAS,KAAz0Z,EAAg1Z,K

AAL,iB;QAA2B,OAAO,I;;MAA5C,UAAU,I;MACV,IAAI,MAAM,sCAAK,UAAX,IAAwB,MAAM,sCAAK,UA
AvC,C;QAAkD,OAAO,I;MACzD,OAAW,OAAJ,GAAI,C;K;IAGf,kC;MAK4C,kCAAsB,EAAtB,C;K;IAE5C,2C;
MAQmB,Q;MAAA,qBAAL,SAAK,EAAY,KAAZ,C;MAAL,iB;QAA2B,OAAO,I;;MAA5C,UAAU,I;MACV,IAAI
,MAAM,uCAAM,UAAZ,IAAyB,MAAM,uCAAM,UAAzC,C;QAAoD,OAAO,I;MAC3D,OAAW,QAAJ,GAAI,C;
K;IAGf,gC;MAKwC,gCAAoB,EAApB,C;K;IAExC,yC;MAQI,WAAW,KAAX,C;MAEA,aAAa,SAAK,O;MAClB,
IAAI,WAAU,CAAd,C;QAAiB,OAAO,I;MAExB,S;MACA,c;MACA,S;MAEA,gBAAgB,qBAAK,CAAL,C;MACh
B,IAAI,YAAY,EAAhB,C;QACI,IAAI,WAAU,CAAd,C;UAAiB,OAAO,I;QAExB,QAAQ,C;QAER,IAAI,cAAa,E
AAjB,C;UACI,aAAa,I;UACb,QAAQ,W;eACL,IAAI,cAAa,EAAjB,C;UACH,aAAa,K;UACb,QAAQ,W;;UAER,O
AAO,I;;QAEX,QAAQ,C;QACR,aAAa,K;QACb,QAAQ,W;;MAIZ,uBAAuB,S;MAEvB,qBAAqB,gB;MACrB,aAA
a,C;MACb,aAAU,KAAV,MAAsB,MAAtB,M;QACI,YAAY,QAAQ,qBAAK,CAAL,CAAR,EAAiB,KAAjB,C;QA
EZ,IAAI,QAAQ,CAAZ,C;UAAe,OAAO,I;QACtB,IAAI,SAAS,cAAb,C;UACI,IAAI,mBAAkB,gBAAtB,C;YACI,i
BAAiB,QAAQ,KAAR,I;YAEjB,IAAI,SAAS,cAAb,C;cACI,OAAO,I;;;YAGX,OAAO,I;;;QAIf,6BAAU,KAAV,C;
QAEA,IAAI,UAAS,QAAQ,KAAR,IAAT,CAAJ,C;UAA4B,OAAO,I;QAEnC,kBAAU,KAAV,I;;MAGJ,OAAW,U
AAJ,GAAgB,MAAhB,GAA4B,CAAC,MAAD,I;K;IAGvC,iC;MAK0C,iCAAqB,EAArB,C;K;IAE1C,0C;MAQI,W
AAW,KAAX,C;MAEA,aAAa,SAAK,O;MAClB,IAAI,WAAU,CAAd,C;QAAiB,OAAO,I;MAExB,S;MACA,c;MA
CA,S;MAEA,gBAAgB,qBAAK,CAAL,C;MAChB,IAAI,YAAY,EAAhB,C;QACI,IAAI,WAAU,CAAd,C;UAAiB,
OAAO,I;QAExB,QAAQ,C;QAER,IAAI,cAAa,EAAjB,C;UACI,aAAa,I;UACb,gC;eACG,IAAI,cAAa,EAAjB,C;U
ACH,aAAa,K;UACb,6B;;UAEA,OAAO,I;;QAEX,QAAQ,C;QACR,aAAa,K;QACb,6B;;MAIJ,2C;MAEA,qBAAq
B,gB;MACrB,e;MACA,aAAU,KAAV,MAAsB,MAAtB,M;QACI,YAAY,QAAQ,qBAAK,CAAL,CAAR,EAAiB,K
AAjB,C;QAEZ,IAAI,QAAQ,CAAZ,C;UAAe,OAAO,I;QACtB,IAAI,uBAAS,cAAT,KAAJ,C;UACI,IAAI,uBAAk
B,gBAAlB,CAAJ,C;YACI,iBAAiB,8BAAQ,KAAR,E;YAEjB,IAAI,uBAAS,cAAT,KAAJ,C;cACI,OAAO,I;;;YAG
X,OAAO,I;;;QAIf,6CAAU,KAAV,E;QAEA,IAAI,uBAAS,8BAAQ,KAAR,EAAT,KAAJ,C;UAA4B,OAAO,I;QAE
nC,6CAAU,KAAV,E;;MAGJ,OAAW,UAAJ,GAAgB,MAAhB,GAA6B,MAAD,a;K;IAIvC,kC;MAAyD,MAAM,0
BAAsB,6BAA0B,KAA1B,MAAtB,C;K;uEwBhI/D,yB;MAAA,oC;MAAA,uC;QAII,iBAAiB,C;QACjB,eAAe,mB
AAS,CAAT,I;QACf,iBAAiB,K;QAEjB,OAAO,cAAc,QAArB,C;UACI,YAAgB,CAAC,UAAL,GAAiB,UAAjB,G
AAiC,Q;UAC7C,YAAY,UAAU,iCAAK,KAAL,EAAV,C;UAEZ,IAAI,CAAC,UAAL,C;YACI,IAAI,CAAC,KAA
L,C;cACI,aAAa,I;;cAEb,0BAAc,CAAd,I;;YAEJ,IAAI,CAAC,KAAL,C;cACI,K;;cAEA,sBAAY,CAAZ,I;;;QAIZ,O
AAO,8BAAY,UAAZ,EAAwB,WAAW,CAAX,IAAxB,C;O;KAzBX,C;yEA4BA,yB;MAAA,8B;MA5BA,oC;MA4
BA,uC;QAIK,Q;QAAsB,kBAAtB,2D;QA5BD,iBAAiB,C;QACjB,eAAe,qBAAS,CAAT,I;QACf,iBAAiB,K;QAEj
B,OAAO,cAAc,QAArB,C;UACI,YAAgB,CAAC,UAAL,GAAiB,UAAjB,GAAiC,Q;UAC7C,YAsBwB,SAtBZ,CA
AU,mCAAK,KAAL,EAAV,C;UAEZ,IAAI,CAAC,UAAL,C;YACI,IAAI,CAAC,KAAL,C;cACI,aAAa,I;;cAEb,0B
AAc,CAAd,I;;YAEJ,IAAI,CAAC,KAAL,C;cACI,K;;cAEA,sBAAY,CAAZ,I;;;QAWZ,OAPO,gCAAY,UAAZ,EA
AwB,WAAW,CAAX,IAAxB,CAOgC,W;O;KAJ3C,C;iFAMA,yB;MAAA,mD;MAAA,oC;MAAA,uC;QAIuB,UA
AL,MAAK,EAAL,MAAK,EAAL,M;QAAK,mBAAL,SAAK,C;QAAL,mB;QAAA,kB;QAAA,kB;QAAd,0D;UACI
,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAV,CAAL,C;YACI,OAAO,8BAAY,KAAZ,EAAmB,gBAAnB,C;QAEf,O
AAO,E;O;KARX,C;mFAWA,yB;MAAA,8B;MAXA,mD;MAAA,oC;MAWA,uC;QAIK,Q;QAAsB,kBAAtB,2D;Q
AAsB,oB;;UAXJ,kC;UAAA,qBAAL,WAAK,C;UAAL,qB;UAAA,oB;UAAA,oB;UAAd,0D;YACI,IAAI,CAUyB,
SAVxB,CAAU,mCAAK,KAAL,EAAV,CAAL,C;cACI,mBAAO,gCAAY,KAAZ,EAAmB,kBAAnB,C;cAAP,qB;;
UAER,mBAAO,E;;;QAOP,OAA4C,2B;O;KAJhD,C;6EAMA,yB;MAAA,mD;MAAA,+C;MAAA,oC;MAAA,uC;Q
AIkB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,CAAQ,CAAb,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,CAA
C,UAAU,iCAAK,KAAL,EAAV,CAAL,C;YACI,OAAO,8BAAY,CAAZ,EAAe,QAAQ,CAAR,IAAf,C;;QAEf,OA
AO,E;O;KARX,C;+EAWA,yB;MAAA,8B;MAXA,mD;MAAA,+C;MAAA,oC;MAWA,uC;QAIK,Q;QAAsB,kBA
AtB,2D;QAAsB,kB;;UAXT,U;UAAA,SAAa,SAAR,YAAL,WAAK,CAAQ,CAAb,W;UAAd,OAAc,gBAAd,C;YA
Ac,yB;YACV,IAAI,CAUuB,SAVtB,CAAU,mCAAK,KAAL,EAAV,CAAL,C;cACI,iBAAO,gCAAY,CAAZ,EAAe
,QAAQ,CAAR,IAAf,C;cAAP,mB;;;UAER,iBAAO,E;;;QAOP,OAA0C,yB;O;KAJ9C,C;IAMA,kC;MAhEI,iBAAAiB
,C;MACjB,eAAe,mBAAS,CAAT,I;MACf,iBAAiB,K;MAEjB,OAAO,cAAc,QAArB,C;QACI,YAAgB,CAAC,UA
AL,GAAiB,UAAjB,GAAiC,Q;QAC7C,YA6DgE,4BA7D1C,iCAAK,KAAL,EA6D0C,E;QA3DhE,IAAI,CAAC,U
AAL,C;UACI,IAAI,CAAC,KAAL,C;YACI,aAAa,I;;YAEb,0BAAc,CAAd,I;;UAEJ,IAAI,CAAC,KAAL,C;YACI,K

;;YAEA,sBAAY,CAAZ,I;;;MAkDiD,OA9CtD,8BAAY,UAAZ,EAAwB,WAAW,CAAX,IAAxB,C;K;IAgDX,kC;MAzCK,Q;MAAsB,kBAAtB,2D;MA5BD,iBAAiB,C;MACjB,eAAe,qBAAS,CAAT,I;MACf,iBAAiB,K;MAEjB,OAAO,cAAc,QAArB,C;QACI,YAAgB,CAAC,UAAL,GAAiB,UAAjB,GAAiC,Q;QAC7C,YAAkEoD,4BAAlE9B,mCAAAK,KAAAL,C;YACI,aAAa,I;;YAEb,0BAAAc,CAAAd,CAAAd,I;;UAEJ,IAAI,CAAC,KAAL,C;YACI,K;;YAEA,sBAAY,CAAZ,I;;;MAuDqC,OAAnD1C,gCAAY,UAAZ,EAAwB,WAAW,CAAX,IAAxB,CAAoC,W;K;IA8C3C,uC;MAGsE,oB;;QA3C/C,C,gC;QAAA,gB;QAAA,mB;QAAA,kB;QAAA,kB;QAAA,kB;QAAA,0D;UAAA,IAAI,CAAC,IAAL,C;YAAA,CAA0CsE,4BAA1C,kBAAAC,KAAAD,C;UAAA,IAAI,CAAC,IAAL,C;YAAA,CAAeE,4BAAf

AAzB,C;K;uFAErE,iC;MAS2E,2BAAY,KAAZ,EAAmB,GAAnB,C;K;mFAE3E,2C;MAO0D,wB;QAAA,WAAgB
,gB;MAAkB,OAAA,8BAAY,UAAZ,EAAwB,QAAxB,CAAkC,W;K;IAE9H,uC;MAG6D,OAAA,8BAAY,KAAM,
MAAlB,EAAyB,KAAM,aAAN,GAAqB,CAArB,IAAzB,CAAiD,W;K;IAE9G,sE;MAImD,qC;QAAA,wBAAgC,S;
MAC/E,YAAY,sBAAQ,SAAR,C;MACZ,OAAW,UAAS,EAApB,GAAwB,qBAAxB,GN1M4F,oBM0M/B,CN1M+
B,EM0M5B,KN1M4B,C;K;IM6MhG,wE;MAIqD,qC;QAAA,wBAAgC,S;MACjF,YAAY,sBAAQ,SAAR,C;MAC
Z,OAAW,UAAS,EAApB,GAAwB,qBAAxB,GNnN4F,oBMmN/B,CNnN+B,EMmN5B,KNnN4B,C;K;IMsNhG,qE
;MAIkD,qC;QAAA,wBAAgC,S;MAC9E,YAAY,sBAAQ,SAAR,C;MACZ,OAAW,UAAS,EAApB,GAAwB,qBA
AxB,GN5N4F,oBM4N/B,QAAQ,CAAR,IN5N+B,EM4NpB,gBN5NoB,C;K;IM+NhG,uE;MAIoD,qC;QAAA,wBA
AgC,S;MAChF,YAAY,sBAAQ,SAAR,C;MACZ,OAAW,UAAS,EAApB,GAAwB,qBAAxB,GNrO4F,oBMqO/B,
QAAQ,SAAU,OAAlB,INrO+B,EMqOL,gBNrOK,C;K;IMwOhG,0E;MAIuD,qC;QAAA,wBAAgC,S;MACnF,YA
AY,0BAAY,SAAZ,C;MACZ,OAAW,UAAS,EAApB,GAAwB,qBAAxB,GN9O4F,oBM8O/B,CN9O+B,EM8O5B,
KN9O4B,C;K;IMiPhG,4E;MAIyD,qC;QAAA,wBAAgC,S;MACrF,YAAY,0BAAY,SAAZ,C;MACZ,OAAW,UAA
S,EApB,GAAwB,qBAAxB,GNvP4F,oBMuP/B,CNvP+B,EMuP5B,KNvP4B,C;K;IM0PhG,yE;MAIsD,qC;QAAA
,wBAAgC,S;MAClF,YAAY,0BAAY,SAAZ,C;MACZ,OAAW,UAAS,EApB,GAAwB,qBAAxB,GNhQ4F,oBMg
Q/B,QAAQ,CAAR,INhQ+B,EMgQpB,gBNhQoB,C;K;IMmQhG,2E;MAIwD,qC;QAAA,wBAAgC,S;MACpF,YA
AY,0BAAY,SAAZ,C;MACZ,OAAW,UAAS,EApB,GAAwB,qBAAxB,GNzQ4F,oBMyQ/B,QAAQ,SAAU,OAA
lB,INzQ+B,EMyQL,gBNzQK,C;K;IM4QhG,oE;MAOI,IAAI,WAAW,UAAf,C;QACI,MAAM,8BAA0B,gBAAa,Q
AAb,oCAAkD,UAAlD,OAA1B,C;MACV,SAAS,sB;MACT,EAAG,qBAAY,SAAZ,EAAkB,CAAlB,EAAqB,UAA
rB,C;MACH,EAAG,gBAAO,WAAP,C;MACH,EAAG,qBAAY,SAAZ,EAAkB,QAAlB,EAA4B,gBAA5B,C;MACH
,OAAO,E;K;yFAGX,yB;MAAA,8B;MAAA,qD;MAAA,+D;QAOK,Q;QAAD,OAAuB,aAAtB,2DAAsB,EAAa,U
Ab,EAAyB,QAAzB,EAAmC,WAAnC,CAAgD,W;O;KAP3E,C;IASA,uD;MAOI,+BAAa,KAAM,MAAnB,EAAyB,
0B,KAAM,aAAN,GAAqB,CAArB,IAA1B,EAAkD,WAAlD,C;K;yFAEJ,yB;MAAA,8B;MAAA,qD;MAAA,gD;Q
AOK,Q;QAAD,OAAuB,aAAtB,2DAAsB,EAAa,KAAb,EAAoB,WAApB,CAAiC,W;O;KAP5D,C;IASA,sD;MASI
,IAAI,WAAW,UAAf,C;QACI,MAAM,8BAA0B,gBAAa,QAAb,oCAAkD,UAAlD,OAA1B,C;MAEV,IAAI,aAAY,
UAAhB,C;QACI,OAAY,mBAAL,SAAK,EAAY,CAAZ,EAAe,gBAAf,C;MAEhB,SAAS,mBAAc,oBAAU,QAAV,
GAAqB,UAArB,C;MACT,EAAG,qBAAY,SAAZ,EAAkB,CAAlB,EAAqB,UAArB,C;MACH,EAAG,gBAAO,WAAP,
EAAY,qBAAZ,EAAkB,QAAlB,EAA4B,gBAA5B,C;MACH,OAAO,E;K;uFAGX,yB;MAAA,8B;MAAA,mD;MAAA,k
D;QASK,Q;QAAD,OAAuB,YAAtB,2DAAsB,EAAY,UAAZ,EAAwB,QAAxB,CAAkC,W;O;KAT7D,C;IAWA,yC
;MAKqE,8BAAY,KAAM,MAAlB,EAAyB,KAAM,aAAN,GAAqB,CAArB,IAAzB,C;K;uFAEjE,yB;MAAA,8B;M
AAA,mD;MAAA,mC;QAOK,Q;QAAD,OAAuB,YAAtB,2DAAsB,EAAY,KAAZ,EAAmB,CAAnB,EAAsB,WAAtB,C
;O;KAP9C,C;IASA,yC;MAKI,IAAI,wBAAW,MAAX,CAAJ,C;QACI,OAAO,8BAAY,MAAO,OAAnB,EAAyB,gBA
A2B,C;;MAEX,OAAO,8BAAY,CAAZ,EAAe,gBAAf,C;MAEhB,IAAI,wBAAW,MAAX,CAAJ,C;QACI,IAAO,
gBAAP,C;MAEJ,IAAI,wBAAW,MAAX,CAAJ,C;QACI,OAAO,8BAAY,MAAO,OAAnB,EAAyB,gBAA3B,C;;
MAAA,OAAO,8BAAY,CAAZ,EAAe,gBAAf,C;K;IAGX,yC;MAKI,IAAI,wBAAW,MAAX,CAAJ,C;QACI,OAAO,
8BAAY,MAAO,OAAnB,EAAyB,gBA2B,C;;MAEX,OAAO,8BAAY,CAAZ,EAAe,gBAAf,C;MAEhB,IAAI,wBA
AW,MAAX,CAAJ,C;QACI,IAAO,gBAAP,C;MAEJ,IAAI,wBAAW,MAAX,CAAJ,C;QACI,OAAO,8BAAY,MAAO,
OAAnB,EAAyB,gBAA3B,C;;MAAA,OAAO,8BAAY,CAAZ,EAAe,gBAAf,C;K;IAGX,yC;MAKI,IAAI,wBAAW,
MAAX,CAAJ,C;QACI,OAAO,8BAAY,MAAO,OAAnB,EAAyB,gBA2B,C;;MAEX,OAAO,8BAAY,CAAZ,EAAe,
gBAAf,C;MAEhB,IAAI,wBAAW,MAAX,CAAJ,C;QACI,IAAO,gBAAP,C;MAEJ,IAAI,wBAAW,MAAX,CAAJ,C;QACI,
OAAO,8BAAY,MAAO,OAAnB,EAAyB,gBA3B,C;;MAAA,OAAO,8BAAY,CAAZ,EAAe,gBAAf,C;K;IAGX,yC
;MAKI,IAAI,wBAAW,MAAX,CAAJ,C;QACI,OAAO,8BAAY,MAAO,OAAnB,EAAyB,gBA2B,C;;MAEX,OAAO,
8BAAY,CAAZ,EAAe,

AtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EA8K4D,WA9K5D,CAAgD,W;;MA8KvE,W;K;IAGJ,kF;MAIuE,qC;QAAA,wBAAgC,S;MACnG,YAAY,sBAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLzF,U;QAAuLM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuL2E,WAAvL3E,CAAgD,W;;MAAuLvE,W;K;IAGJ,oF;MAI2E,qC;QAAA,wBAAgC,S;MACvG,YAAY,0BAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMzF,U;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,sF;MAIyE,qC;QAAA,wBAAgC,S;MACrG,YAAY,4BAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM0E,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,uF;MAI0E,qC;QAAA,wBAAgC,S;MACtG,YAAY,8BAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM4E,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,wF;MAI2E,qC;QAAA,wBAAgC,S;MACvG,YAAY,gCAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM8E,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,yF;MAI4E,qC;QAAA,wBAAgC,S;MACxG,YAAY,kCAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMgF,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,0F;MAI6E,qC;QAAA,wBAAgC,S;MACzG,YAAY,oCAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMkF,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,2F;MAI8E,qC;QAAA,wBAAgC,S;MAC1G,YAAY,sCAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMoF,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,4F;MAI+E,qC;QAAA,wBAAgC,S;MAC3G,YAAY,wCAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMsF,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,6F;MAIgF,qC;QAAA,wBAAgC,S;MAC5G,YAAY,0CAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMwF,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,8F;MAIiF,qC;QAAA,wBAAgC,S;MAC7G,YAAY,4CAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM0F,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,+F;MAIkF,qC;QAAA,wBAAgC,S;MAC9G,YAAY,8CAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM4F,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,gG;MAImF,qC;QAAA,wBAAgC,S;MAC/G,YAAY,gDAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM8F,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,iG;MAIoF,qC;QAAA,wBAAgC,S;MAChH,YAAY,kDAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMgG,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,kG;MAIqF,qC;QAAA,wBAAgC,S;MACjH,YAAY,oDAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMkG,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,mG;MAIsF,qC;QAAA,wBAAgC,S;MAClH,YAAY,sDAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMoG,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,oG;MAIuF,qC;QAAA,wBAAgC,S;MACnH,YAAY,wDAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMsG,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,qG;MAIwF,qC;QAAA,wBAAgC,S;MACpH,YAAY,0DAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMwG,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,sG;MAIyF,qC;QAAA,wBAAgC,S;MACrH,YAAY,4DAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM0G,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,uG;MAI0F,qC;QAAA,wBAAgC,S;MACtH,YAAY,8DAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM4G,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,wG;MAI2F,qC;QAAA,wBAAgC,S;MACvH,YAAY,gEAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM8G,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,yG;MAI4F,qC;QAAA,wBAAgC,S;MACxH,YAAY,kEAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMgH,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,0G;MAI6F,qC;QAAA,wBAAgC,S;MACzH,YAAY,oEAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMkH,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,2G;MAI8F,qC;QAAA,wBAAgC,S;MAC1H,YAAY,sEAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMoH,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,4G;MAI+F,qC;QAAA,wBAAgC,S;MAC3H,YAAY,wEAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMsH,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,6G;MAIgG,qC;QAAA,wBAAgC,S;MAC5H,YAAY,0EAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMwH,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,8G;MAIiG,qC;QAAA,wBAAgC,S;MAC7H,YAAY,4EAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM0H,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,+G;MAIkG,qC;QAAA,wBAAgC,S;MAC9H,YAAY,8EAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM4H,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,gI;MAImG,qC;QAAA,wBAAgC,S;MAC/H,YAAY,gFAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM8H,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,iI;MAIoG,qC;QAAA,wBAAgC,S;MAChI,YAAY,kFAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMgI,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,kI;MAIqG,qC;QAAA,wBAAgC,S;MACjI,YAAY,oFAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMkI,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,mI;MAIsG,qC;QAAA,wBAAgC,S;MAClI,YAAY,sFAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMoI,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,oI;MAIuG,qC;QAAA,wBAAgC,S;MACnI,YAAY,wFAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMsI,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,qI;MAIwG,qC;QAAA,wBAAgC,S;MACpI,YAAY,0FAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMwI,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,sI;MAIyG,qC;QAAA,wBAAgC,S;MACrI,YAAY,4FAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM0I,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,uI;MAI0G,qC;QAAA,wBAAgC,S;MACtI,YAAY,8FAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM4I,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,wI;MAI2G,qC;QAAA,wBAAgC,S;MACvI,YAAY,gGAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM8I,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,yI;MAI4G,qC;QAAA,wBAAgC,S;MACxI,YAAY,kGAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMgJ,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,0I;MAI6G,qC;QAAA,wBAAgC,S;MACzI,YAAY,oGAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMkJ,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,2I;MAI8G,qC;QAAA,wBAAgC,S;MAC1I,YAAY,sGAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMoJ,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,4I;MAI+G,qC;QAAA,wBAAgC,S;MAC3I,YAAY,wGAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMsJ,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,6I;MAIgH,qC;QAAA,wBAAgC,S;MAC5I,YAAY,0GAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMwJ,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,8I;MAIiH,qC;QAAA,wBAAgC,S;MAC7I,YAAY,4GAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM0J,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,+I;MAIkH,qC;QAAA,wBAAgC,S;MAC9I,YAAY,8GAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM4J,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,gK;MAImH,qC;QAAA,wBAAgC,S;MAC/I,YAAY,gHAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM8J,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,iK;MAIoH,qC;QAAA,wBAAgC,S;MAChJ,YAAY,kHAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMgK,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,kK;MAIqH,qC;QAAA,wBAAgC,S;MACjJ,YAAY,oHAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMkK,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,mK;MAIsH,qC;QAAA,wBAAgC,S;MAClJ,YAAY,sHAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMoK,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,oK;MAIuH,qC;QAAA,wBAAgC,S;MACnJ,YAAY,wHAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMsK,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,qK;MAIwH,qC;QAAA,wBAAgC,S;MACpJ,YAAY,0HAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMwK,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,sK;MAIyH,qC;QAAA,wBAAgC,S;MACrJ,YAAY,4HAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM0K,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,uK;MAI0H,qC;QAAA,wBAAgC,S;MACtJ,YAAY,8HAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM4K,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,wK;MAI2H,qC;QAAA,wBAAgC,S;MACvJ,YAAY,gIAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM8K,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,yK;MAI4H,qC;QAAA,wBAAgC,S;MACxJ,YAAY,kIAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMgL,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,0K;MAI6H,qC;QAAA,wBAAgC,S;MACzJ,YAAY,oIAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMkL,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,2K;MAI8H,qC;QAAA,wBAAgC,S;MAC1J,YAAY,sIAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMoL,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,4K;MAI+H,qC;QAAA,wBAAgC,S;MAC3J,YAAY,wIAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMsL,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,6K;MAIgI,qC;QAAA,wBAAgC,S;MAC5J,YAAY,0IAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMwL,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,8K;MAIiI,qC;QAAA,wBAAgC,S;MAC7J,YAAY,4IAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM0L,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,+K;MAIkI,qC;QAAA,wBAAgC,S;MAC9J,YAAY,8IAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM4L,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,gM;MAImI,qC;QAAA,wBAAgC,S;MAC/J,YAAY,gJAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hM8L,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,iM;MAIoI,qC;QAAA,wBAAgC,S;MAChK,YAAY,kJAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMgM,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,kM;MAIqI,qC;QAAA,wBAAgC,S;MACjK,YAAY,oJAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMkM,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAAuLgE,WAAvLhE,CAAgD,W;;MAAuLvE,W;K;IAGJ,mM;MAIsI,qC;QAAA,wBAAgC,S;MAClK,YAAY,sJAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAlB,I;QAAb,eAAuC,gB;QAAvLhM,hMoM,qC;QAAuLiM,OAAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,Q

CikXR,kB;YAAgB,cAAhB,UiCjkXQ,KjCikXR,S;YAAsB,IiCjkXC,SAAH,UjCikXgB,oBiCjkXhB,CAAG,0BjCikXD,C;cAAwB,aAAO,I;cAAP,e;;;UAC9C,aAAO,K;;;QiClkXH,e;UACI,OAAO,K;;MAEf,OAAO,E;K;IAGX,kE;MASyD,0B;QAAA,aAAkB,2B;MAAW,0B;QAAA,aAAsB,K;MACxG,IAAI,CAAC,UAAD,IAAe,EAAa,KAAM,OAAN,K;AAc,CAA7B,IAAkC,6BAAtC,C;QACI,WAAiB,SAAN,KAAM,C;QACjB,ONruB4F,sB8G3MM,oBxGg7BzE,IwGh7ByE,C9G2MN,EMquB7D,UNruB6D,C;;kBMyuBhG,iBAAyB,eAAX,UAAW,EAAa,2BAAb,CAAzB,WAAwD,CAAxD,U;QACI,kBAAkB,qBAAI,KAAJ,C;QACR,c;;UjCyiXE,Q;UAAhB,wBiCziXQ,KjCyiXR,gB;YAAgB,cAAhB,UiCziXQ,KjCyiXR,O;YAAsB,IiCziXC,SAAH,UjCyiXgB,oBiCziXhB,CAAG,0BjCyiXD,C;cAAwB,aAAO,I;cAAP,e;;;UAC9C,aAAO,K;;;QiC1iXH,e;UACI,OAAO,K;;MAGf,OAAO,E;K;IAIX,8E;MAA2G,oB;QAAA,OAAgB,K;MAOrH,OrG,UAKA,M;MAXlB,cAAkB,CAAC,IAAL,GAAK,CAAV,aAAW,gBAAX,UAAW,EAAc,CAAd,CAAX,EAAsC,eAAT,QAAS,EAAa,EAAb,CAAb,gBAAa,UAAW,EAAc,2BAAd,CAAX,EAAmD,gBAAT,QAAS,EAAc,CAAd,CAAc,CAAnD,C;MAEJ,IAAI,iCAAkB,yBAAT,tB,C;QAAS,kB,yB;QAAd,Ad,OAAc,cAAd,C;UAAc,uB;UAAd,VAAc,IAAd,cAAc,EAAa,EAAb,CAAb,EAAsC,eAAT,QAAS,EAAa,EAAb,CAAb,gBAAa,UAAW,EAAc,2BAAd,CAAX,EAAmD,gBAAT,QAAS,EAAc,CAAd,CAAc,CAAnD,C;QAAS,OAAT,C;YAAS,IAAT,C;;;

AkB,E;;UAElB,YAAkB,iDAAN,yCAAM,EAAa,oBAAb,C;UAClB,IAAI,SAAS,IAAb,C;YACI,gBAAW,qCAAyB,iBAAN,yCAAM,CAAzB,C;YACX,uBAAkB,E;;YAElB,IAAK,QAAiB,KAAjB,aAAL,EAAY,SAAU,KAAV,a;YACZ,gBAAW,gCAAwB,KAAxB,C;YACX,yBAAoB,QAAQ,MAAR,I;YACpB,uBAAkB,0BAAwB,WAAU,CAAd,GAAiB,CAAjB,GAAwB,CAA5C,K;;;QAG1B,iBAAY,C;;K;oEAIpB,Y;MAKiB,Q;MAJb,IAAI,mBAAa,EAAjB,C;;QACI,iB;MACJ,IAAI,mBAAa,CAAjB,C;QACI,MAAM,6B;MACV,aAAa,mE;MAEb,gBAAW,I;MACX,iBAAY,E;MACZ,OAAO,M;K;uEAGX,Y;MACI,IAAI,mBAAa,EAAjB,C;QACI,iB;MACJ,OAAO,mBAAa,C;K;;iDA9C5B,Y;MAA8C,+D;K;;IAgEU,0E;MAAA,0C;QAhBpD,hB1mCjD,SgB2mCH,sBAAW,kBAAX,EAAuB,YAAvB,EAAkD,kBAAlD,C;QAAA,OAAA,wE,YAAA,C;QAAA,OAAA,wE,YAAA,C;QAAA,OAAA,wE,YAAA

C,WAC6B,e;MAC7B,YAC8B,e;MAC9B,eACiC,e;MACjC,YAC8B,gB;MAC9B,aAC+B,gB;MAC/B,YAC8B,gB;MAC9B,aAC+B,gB;MAC/B,eACiC,gB;MACjC,iBACmC,gB;MACnC,qBAEuC,gB;MACvC,sBAEwC,gB;MACxC,kBACoC,gB;MACpC,cACgC,gB;MAChC,iBACmC,gB;MACnC,iBACmC,gB;MACnC,iBACmC,gB;MACnC,YAC8B,gB;MAC9B,aAC+B,iB;MAC/B,aAC+B,iB;MAC/B,uBACyC,iB;MACzC,wBAC0C,iB;MAC1C,sBACwC,iB;MACxC,uBACyC,iB;MACzC,wBAC0C,iB;MAC1C,sBACwC,iB;MACxC,cACgC,iB;MAChC,oBACsC,iB;MACtC,cACgC,iB;MAChC,gBACkC,iB;MAClC,aAC+B,iB;MAC/B,mBACqC,iB;MACrC,YAC8B,iB;MAC9B,UAC4B,iB;MAC5B,mBACqC,iB;MACrC,gBACkC,iB;MAClC,mBACqC,iB;MACrC,sBACwC,iB;MAExC,sBAGwC,gB;MAExC,uBAGyC,gB;K;;;IA7F7C,kC;MAAA,iC;QAAA,gB;;MAAA,0B;K;;;;;;;;;2FCuE0C,Y;MAAQ,oCAAa,IAAb,C;K;IAAiBpB,yC;MAAqB,kB;K;mIAC3C,Y;MACmD,OAAA,UAAM,YAAN,aAAkB,CAAlB,C;K;mIACnD

QACF,+C;UACE,MAAM,6BAAyB,sCAAmC,KAAnC,OAAzB,EAAsE,CAAtE,C;;UAHkC,O;;K;0DAM5C,iB;;QAMqD,OACjD,cAAc,KAAd,EAAiC,IAAjC,C;;QACF,+C;UACE,MAAM,6BAAyB,0CAAuC,KAAvC,OAAzB,EAA0E,CAA1E,C;;UAH2C,O;;K;uDAMrD,iB;;QAWmD,OAC/C,cAAc,KAAd,EAAiC,KAAjC,C;;QACF,+C;UAFiD,OAG/C,I;;UAH+C,O;;K;gEAMnD,iB;;QAK4D,OACxD,cAAc,KAAd,EAAiC,IAAjC,C;;QACF,+C;UAF0D,OAGxD,I;;UAHwD,O;;K;;;IA/XhE,0C;MAAA,yC;QAAA,wB;;MAAA,kC;K;oCAwYA,Y;MAC6C,kBAAY,YAAD,aAAAX,EAAzB,KAAyB,QAAAT,GAAA,qBAAAC,GAAA,qBAAAC,GAAA,qBAAAC

UAAR,EAAuB,Q;K;0CAMxC,gB;MAQiB,UAAN,M;MAAM,sB;MACT,iBAAA,yCAAS,WAAT,E;QAA4B,SAAP,wCAAO,kB;WAC5B,iBAAA,qDAAa,WAAb,E;QAAgC,SAAP,wCAAO,kB;;QAG5B,6BAAoB,YAAM,WA1BEAAsC,kBAAAtC,EAAAmD,IAAAnD,C;;MALR,a;K;wCAUJ,gB;MAUiB,UAAN,M;MAAM,sB;MACT,iBAAA,yCAAS,WAAT,E;;WACA,iBAAA,qDAAa,WAAb,E;;;QACQ,+BAAoB,YAAApB,EAAA2B,kBAAAvB,EAAAwB,IAAAxB,C;MAHZ,a;K;uCAOJ,gB;MAUI,OAAAa,WAAb,oBAAAO,EAAAC,IAAAD,C;MAAa,a;K;kFAAb,Y;MAAa,6D;K;mFAAb,Y;MAAa,8D;K;qFAAN,Y;MAAM,gE;K;qFAAR,Y;MAAQ,gE;K;0FAAH,Y;MAAG,qE;K;0FAAR,Y;MAAQ,qE;K;yFAAT,Y;MAAS,oE;K;uFASrC,Y;MAAqC,2D;K;wFAAR,Y;MAAQ,4D;K;0FAAR,Y;MAAQ,8D;K;0FAAR,Y;MAAQ,8D;K;+FAAR,Y;MACI,OAAAW,uBAAAgB,eAAApB,EAAAqB,GAAAC,YAAAhC,C,GAAA2B,kBAAAM,EAAAC,IAAAD,C;+FAAtB,Y;MAAsB,mE;K;8FAAR,Y;MAAQ,wEAAR,Q;MAAQ,gE

AU,C;MACf,IAAI,sEAAqB,SAArB,CAAJ,C;QACH,uBAAgB,KAAhB,C;;QAEA,aAAwE,YAA3D,oBAAoB,SAApB,EAA0B,IAA1B,B,0CAA2D,C;QACxE,kCAA2B,MAA3B,C;;MAJJ,W;K;IAgBuB,oC;MAAQ,oE;K;IAOP,sC;MAAQ,sE;K;IAWN,sC;MAAQ,sE;K;IAQV,qC;MAAQ,qE;K;IAOP,uC;MAAQ,uE;K;IAWN,uC;MAAQ,uE;K;IAQX,qC;MAAQ,qE;K;IAOP,uC;MAAQ,uE;K;IAWN,uC;MAAQ,uE;K;IAQhB,gC;MAAQ,gE;K;IAOP,kC;MAAQ,kE;K;IAWN,kC;MAAQ,kE;K;IAQX,gC;MAAQ,gE;K;IAOP,kC;MAAQ,kE;K;IAWN,kC;MAAQ,kE;K;IAQb,8B;MAAQ,8D;K;IAOP,gC;MAAQ,gE;K;IAWN,gC;MAAQ,gE;K;IAQZ,6B;MAAQ,6D;K;IAOP,+B;MAAQ,+D;K;IAWN,+B;MAAQ,+D;K;yEAG/B,+B;MAIqE,8BAAW,SAAX,C;K;2EAErE,+B;MAUwE,8BAAW,SAAX,C;K;IAIxE,yC;MACI,aAAa,KAAM,O;MACnB,IAAI,WAAU,CAAd,C;QAAiB,MAAM,gCAAyB,qBAAzB,C;MACvB,YAAY,C;MACZ,aAAa,gCAAS,K;MACtB,qBAAqB,U;MACrB,QAAM,iBAAM,KAAN,CAAN,C;aACI,E;aAAA,E;UAAY,qB;UAAZ,K;;MAEJ,cAAc,QAAQ,C;MACtB,iBAAiB,WAAiB,aAAN,KAAM,EAAW,EAAX,C;MAE9B,cAAU,KAAV,EAAgB,C;QAAC,gCAAS,eAAA,EAAT,C;MAEjB

C,EU4ThD,C;cAAyB,aAAO,K;cAAP,e;;;UAC/C,aAAO,I;;;QV7TyD,iB;;MAAhE,S;QAEI,OAAW,iBAAM,CAAN
,MAAY,EAAhB,sD;;MAGX,OAAiB,WAAN,KAAM,EAAW,GAAX,CAAV,GAAyC,OAAR,QAAN,KAAM,EAA
K,CAAL,CAAQ,CAAzC,GAA6D,OAAN,KAAM,C;K;IAKxE,0D;MAII,QAHgC,U;MAIhC,OAAO,IAAI,gBAAJ,I
AJqC,SAIvB,CAAU,iCAAK,CAAL,EAAV,CAArB,C;QAAyC,a;;MAJzC,OiBnqC4F,oBjBmqClF,UiBnqCkF,EjB
wqCrF,CiBxqCqF,C;K;IjBqqChG,qD;MACI,QAAQ,U;MACR,OAAO,IAAI,gBAAJ,IAAc,UAAU,iCAAK,CAAL,
EAAV,CAArB,C;QAAyC,a;;MACzC,OAAO,C;K;;;;;IAmBX,8B;MAA+C,qCAAQ,OAAR,E;K;IAC/C,+B;MAAg
D,2CAAS,OAAT,E;K;IAEhD,sC;MAAiD,oBAAS,sBAAgB,CAAhB,CAAT,C;K;IACjD,wC;MAAmD,oBAAU,uB
AAiB,CAAjB,CAAD,yBAAuB,CAAvB,EAAT,C;K;IACnD,oD;MAAoE,oBAAU,sBAAgB,CAAhB,CAAD,yBAA
sB,iBAAtB,EAAT,C;K;IACpE,0C;MACI,IAAI,sEAAqB,SAArB,CAAJ,C;QAAA,OACI,gBAAgB,KAAhB,C;;QA
DJ,OAGI,iBAAiB,cAAc,KAAd,CAAjB,C;;K;IAGR,4C;MACI,IAAI,kEAAgC,mBAAhC,CAAJ,C;QAAA,OACI,g
BAAgB,cAAc,MAAd,CAAhB,C;;QADJ,OAGI,iBAAwB,WAAP,MAAO,yBAAsB,UAAtB,CAAxB,C;;K;IwMl3C
R,8B;MAEgD,QAAM,SAAN,M;aAC5C,a;UAD4C,OAChB,I;aAC5B,c;UAF4C,OAEf,I;aAC7B,c;UAH4C,OAGf,I;
aAC7B,S;UAJ4C,OAIpB,G;aACxB,S;UAL4C,OAKpB,G;aACxB,O;UAN4C,OAMtB,G;aACtB,M;UAP4C,OAOv
B,G;;UpMuEwB,MAAM,6BAA8B,CoMtEnE,mBAAgB,SpMsEmD,YAA9B,C;;K;IoMnEvD,4C;MACwE,QAAM,
SAAN,C;aACpE,I;UADoE,6C;aAEpE,I;UAFoE,8C;aAGpE,I;UAHoE,8C;aAIpE,G;UAJoE,yC;aAKpE,G;UALoE,y
C;aAMpE,G;UANoE,uC;aAOpE,G;UAPoE,sC;;UAQ5D,MAAM,gCAAyB,uCAAoC,SAA7D,C;;K;IAGlB,yD;MA
GQ,KAAC,eAAD,C;QAEQ,IADE,OACF,Q;UAHZ,sC;;UAIoB,MAAM,gCAAyB,4EAAqD,OAArD,CAAzB,C;;Q
AIlB,QAAM,OAAN,C;eACI,E;YATZ,uC;eAUY,E;YAVZ,yC;eAWY,E;YAXZ,yC;;YAYoB,MAAM,gCAAyB,yD
AAkC,OAAlC,CAAzB,C;;;K;IC5F9B,4B;K;;;MC4BI,kC;;IAXA,gC;MAAA,oC;MAM0B,2BAAc,iC;K;8CACpC,Y
;MAAkC,OAAA,iCAAoB,W;K;6CADhC,Y;MAAA,yC;K;;;IAN1B,4C;MAAA,2C;QAAA,0B;;MAAA,oC;K;IAW
A,gC;MAAA,oC;K;;;IAAA,4C;MAAA,2C;QAAA,0B;;MAAA,oC;K;;IAKJ,oB;K;qCAcI,oB;MAK8D,4BAAiB,IA
AjB,EAAuB,QAAvB,C;K;sCAE9D,oB;MAK+D,wBAAM,QAAD,aAAL,C;K;sCAG/D,Y;MAMqC,QAAC,iBAAa,a,
a;K;yCAEnD,Y;MAMwC,OAAA,iBAAa,a;K;;4EAIzD,yB;MAAA,4C;MAAA,mC;QAQuE,MAAM,WAAM,0BA
AN,C;O;KAR7E,C;mFAUA,yB;MAAA,4C;MAAA,mC;QAQsE,MAAM,WAAM,0BAAN,C;O;KAR5E,C;IAY8B,
4C;MAAiD,mB;MAAhD,gB;MAAoB,4B;K;4CAC/C,Y;MAAsC,OAAA,SAAK,aAAL,cAAoB,eAApB,C;K;6CAEt
C,oB;MAAkD,4BAAiB,SAAjB,EAAuB,4BAAa,QAAb,CAAvB,C;K;;IChGV,sC;MAAC,gB;K;IAOf,4E;MAA8G,
mB;MAA7G,4B;MAA6B,8B;MAAgD,sB;K;+DACpG,Y;MAAsC,OAAA,gC,aAA/B,iBAAW,OAAX,UAAoB,gBAAa
pB,CAA+B,EAAW,iBAAW,KAAtB,CAAhC,cAA8D,aAA9D,C;K;gECtC,oB;MAAkD,+CAAQ,aAAR,E;K;+CAAQ,a
,iBAAxB,EAAoC,0BAAS,QAAT,CAApC,C;K;;+CAGtD,Y;MAAmC,+CAAQ,aAAR,WAAb,EAAqB,IAArB,EAA2B,gCA
AS,KAApC,C;K;;IAUO,wC;MAAC,gB;K;IAOf,gF;MAA8G,mB;MAA7G,4B;MAA6B,8B;MAAgD,sB;K;mEAC1
G,Y;MAAsC,OAAA,gC,aAA/B,iBAAW,OAAX,UAAoB,gBAAa,iBAAW,OAAX,UAAoB,gBAAa,iBAAW,KAAtB,CAAhC,cAA8D,aA9D
,C;K;oECtC,oB;MAAkD,mDAAQ,aAAR,E;K;mDAAQ,a,iBAAxB,EAAoC,0BAAS,QAAT,CAApC,C;K;;iDAGtD,Y;M
AAmC,mDAAQ,aAAR,WAAb,EAAqB,IAArB,EAA2B,gCAA

GtC,6B;MAEmD,UAMX,M;MAPxC,kBACmD,mE;MAEnD,eAC0B,K;MAE1B,cACwC,kE;MAExC,gBACmC,gB;K;iGAG/B,Y;MAAQ,0C;K;0DAEZ,kB;MACI,cAAY,I;MACZ,gBAAc,M;K;IAGsE,iG;MAAA,uB;QAExE,Q;QAAZ,qCAAY,8D;QACZ,sCAAa,a;QAFb,OAGA,yB;O;K;2DAJJ,+B;MAAkD,OAAsC,wDAAtC,c;K;IAOyE,uH;MAAA,uB;QAExG,Q;QAAf,iBAAe,8F;QACf,eAAK,2B;QAA6B,mC;QtMjGtB,gBAAT,Q;QsMsG0D,kB;QAJzD,sBAAsB,SAAK,W;QAC3B,IAAI,eAAa,eAAjB,C;UAEI,iC;UACA,mBAAY,oCAAwB,eAAxB,EAAyC,kEAAzC,C,C;;UAGZ,mBAAY,kE;;QAEhB,oBAAa,e;QAAb,yB;O;K;6DAAJ,0C;MAAqF,OAAsC,qEAAtC,c;K;IAAqB,zB,mI;MAAA,qB;QACxH,yCAAgB,uB;QAAhB,qCAAY,Y;QAAZ,uCAAc,E;QAAd,lB,W;O;K;iEATA,iC;MAUI,wCAAa,mCAAb,EAAoC,kFAApC,C,C;K;mDAAJ,xB,Y;MAAuB,OAAsC,uDAAtC,c,C;;K;;0EAAvB,mB,4B;MAAA,oI;QAAA,oB;QAAA,qB;QAAA,oB;QAAA,qB;QAAA,oB;QAAA,qB

B;MAoBI,QAhBA,SAgBQ,GAhBY,KAgBZ,I;MACR,IAAI,CAjBJ,SAiBI,GAjBgB,KAiBhB,IAAiB,CAAjB,IAAsB,eAjBN,KAiBM,MAjB1B,SAiBA,C;QAA6C,a;;MAjB7C,OAkBO,C;K;mEAhBX,yB;MAAA,4B;MAAA,mC;QA2BI,QAnBA,SAmBQ,GAnBO,K;QAAf,OAAwB,QAoBjB,KApBQ,KAoBH,GAAW,CAAC,CAAC,IApBV,KAoBS,KAAmB,KAAK,CAAC,CAAD,IAAL,CAAnB,CAAD,KAAkC,EAAlD,KApBiB,C;O;KAR5B,C;4EAUA,4B;MAII,QAAQ,YAAO,KAAP,I;MACR,IAAI,aAAS,KAAT,IAAiB,CAAjB,IAAsB,eAAI,KAAJ,MAAa,SAAvC,C;QAA6C,a;;MAC7C,OAAO,C;K;mEAAX,4B;MAAA,QAAI,QAAQ,YAAO,K;MAAP

IAAK,MAAL,KAAc,KAAd,IACf,IAAK,MAAL,IAAc,KADrB,CADc,CAAvB,C;K;IAIJ,mC;MAAA,uC;MACI,2BAIuC,G;MAEvC,eAIoC,uCAA0B,M;K;;;IAXlE,+C;MAAA,8C;QAAA,6B;;MAAA,uC;K;;IA9CA,iD;MAAA,uD;MAG6C,0BAAK,KAAL,EAAY,KAAZ,EAAmB,CAAnB,C;MAH7C,Y;K;IA6DJ,qC;MAAA,yC;K;8CAEI,Y;MAC2B,yBAAc,CAAd,EAAiB,CAAjB,EAAoB,EAApB,C;K;;;IAH/B,iD;MAAA,gD;QAAA,+B;;MAAA,yC;K;4FCxDI,yB;MAAA,2D;MAAA,4B;QAAQ,MAAM,6BAAoB,6BAApB,C;O;KAAd,C;;;;ICSJ,uB;MAG2C,+BAAoB,KAApB,C;K;4EAE3C,wC;MAO4F,sB;K;IAE5F,6C;MAAA,e;MAAA,iB;MAAA,uB;K;IAAA,2C;MAAA,8C;O;MAKI,wF;MAKA,sF;MAMA,wE;K;;IAXA,yD;MAAA,iC;MAAA,iD;K;;IAKA,wD;MAAA,iC;MAAA,gD;K;;IAMA,iD;MAAA,iC;MAAA,yC;K;;IAhBJ,uC;MAAA,iJ;K;;IAAA,4C;MAAA,a;aAAA,c;UAAA,sD;aAAA,a;UAAA,qD;aAAA,M;UAAA,8C;;;UAAA,gE;;;K;;IAyBA,+B,+B;MAAA,mC;K;;;IAAA,2C;MAAA,0C;QAAA,yB;;MAAA,mC;K;IAGoC,qC;MAAA,uC;MAAA,qB;MAAA,wC;MAAA,gB;MAAA,2B,iC;K;uFAGvB,Y;MAAA,wC,Q;MALP,IAAI,kBAAW,iCAAf,C;QAAI,gBAAS,mC;QAAT,qCACI,qBAAc,I;;QADlB,qCAEI,qBAAc,I;;QAFlB,qCAGI,qBAAc,I;;QAHlB,qCAII,qBAAc,I;;QAJlB,qCAKI,qBAAc,I;;QALlB,qCAMI,qBAAc,I;;QANlB,qCAOI,qBAAc,I;;QAPlB,qCAQI,qBAAc,I;;QARlB,qCASI,qBAAc,I;;QATlB,qCAUI,qBAAc,I;;QAVlB,qCAWI,qBAAc,I;;QAXlB,qCAYI,qBAAc,I;;QAZlB,qCAaI,qBAAc,I;;QAblB,qCAcI,qBAAc,I;;QAdlB,qCAeI,qBAAc,I;;QAflB,qCAgBI,qBAAc,I;;QAhBlB,qCAiBI,qBAAc,I;;QAjBlB,qCAkBI,qBAAc,I;;QAlBlB,qCAmBI,qBAAc,I;;QAnBlB,qCAoBI,qBAAc,I;;QApBlB,qCAqBI,qBAAc,I;;QArBlB,qCAsBI,qBAAc,I;;QAtBlB,qCAuBI,qBAAc,I;;QAvBlB,qCAwBI,qBAAc,I;;QAxBlB,qCAyBI,qBAAc,I;;QAzBlB,qCA0BI,qBAAc,I;;QA1BlB,qCA2BI,qBAAc,I;;QA3BlB,qCA4BI,qBAAc,I;;QA5BlB,qCA6BI,qBAAc,I;;QA7BlB,qCA8BI,qBAAc,I;;QA9BlB,qCA+BI,qBAAc,I;;QA/BlB,qCAgCI,qBAAc,I;;QAhClB,qCAiCI,qBAAc,I;;QAjClB,qCAkCI,qBAAc,I;;QAlClB,qCAmCI,qBAAc,I;;QAnClB,qCAoCI,qBAAc,I;;QApClB,qCAqCI,qBAAc,I;;QArClB,qCAsCI,qBAAc,I;;QAtClB,qCAuCI,qBAAc,I;;QAvClB,qCAwCI,qBAAc,I;;QAxClB,qCAyCI,qBAAc,I;;QAzClB,qCA0CI,qBAAc,I;;QA1ClB,qCA2CI,qBAAc,I;;QA3ClB,qCA4CI,qBAAc,I;;QA5ClB,qCA6CI,qBAAc,I;;QA7ClB,qCA8CI,qBAAc,I;;QA9ClB,qCA+CI,qBAAc,I;;QA/ClB,qCAgDI,qBAAc,I;;QAhDlB,qCAiDI,qBAAc,I;;QAjDlB,qCAkDI,qBAAc,I;;QAlDlB,qCAmDI,qBAAc,I;;QAnDlB,qCAoDI,qBAAc,I;;QApDlB,qCAqDI,qBAAc,I;;QArDlB,qCAsDI,qBAAc,I;;QAtDlB,qCAuDI,qBAAc,I;;QAvDlB,qCAwDI,qBAAc,I;;QAxDlB,qCAyDI,qBAAc,I;;QAzDlB,qCA0DI,qBAAc,I;;QA1DlB,qCA2DI,qBAAc,I;;QA3DlB,qCA4DI,qBAAc,I;;QA5DlB,qCA6DI,qBAAc,I;;QA7DlB,qCA8DI,qBAAc,I;;QA9DlB,qCA+DI,qBAAc,I;;QA/DlB,qCAgEI,qBAAc,I;;QAhElB,qCAiEI,qBAAc,I;;QAjElB,qCAkEI,qBAAc,I;;QAlElB,qCAmEI,qBAAc,I;;QAnElB,qCAoEI,qBAAc,I;;QApElB,qCAqEI,qBAAc,I;;QArElB,qCAsEI,qBAAc,I;;QAtElB,qCAuEI,qBAAc,I;;QAvElB,qCAwEI,qBAAc,I;;QAxElB,qCAyEI,qBAAc,I;;QAzElB,qCA0EI,qBAAc,I;;QA1ElB,qCA2EI,qBAAc,I;;QA3ElB,qCA4EI,qBAAc,I;;QA5ElB,qCA6EI,qBAAc,I;;QA7ElB,qCA8EI,qBAAc,I;;QA9ElB,qCA+EI,qBAAc,I;;QA/ElB,qCAgFI,qBAAc,I;;QAhFlB,qCAiFI,qBAAc,I;;QAjFlB,qCAkFI,qBAAc,I;;QAlFlB,qCAmFI,qBAAc,I;;QAnFlB,qCAoFI,qBAAc,I;;QApFlB,qCAqFI,qBAAc,I;;QArFlB,qCAsFI,qBAAc,I;;QAtFlB,qCAuFI,qBAAc,I;;QAvFlB,qCAwFI,qBAAc,I;;QAxFlB,qCAyFI,qBAAc,I;;QAzFlB,qCA0FI,qBAAc,I;;QA1FlB,qCA2FI,qBAAc,I;;QA3FlB,qCA4FI,qBAAc,I;;QA5FlB,qCA6FI,qBAAc,I;;QA7FlB,qCA8FI,qBAAc,I;;QA9FlB,qCA+FI,qBAAc,I;;QA/FlB,qCAgGI,qBAAc,I;;QAhGlB,qCAiGI,qBAAc,I;;QAjGlB,qCAkGI,qBAAc,I;;QAlGlB,qCAmGI,qBAAc,I;;QAnGlB,qCAoGI,qBAAc,I;;QApGlB,qCAqGI,qBAAc,I;;QArGlB,qCAsGI,qBAAc,I;;QAtGlB,qCAuGI,qBAAc,I;;QAvGlB,qCAwGI,qBAAc,I;;QAxGlB,qCAyGI,qBAAc,I;;QAzGlB,qCA0GI,qBAAc,I;;QA1GlB,qCA2GI,qBAAc,I;;QA3GlB,qCA4GI,qBAAc,I;;QA5GlB,qCA6GI,qBAAc,I;;QA7GlB,qCA8GI,qBAAc,I;;QA9GlB,qCA+GI,qBAAc,I;;QA/GlB,qCAgHI,qBAAc,I;;QAhHlB,qCAiHI,qBAAc,I;;QAjHlB,qCAkHI,qBAAc,I;;QAlHlB,qCAmHI,qBAAc,I;;QAnHlB,qCAoHI,qBAAc,I;;QApHlB,qCAqHI,qBAAc,I;;QArHlB,qCAsHI,qBAAc,I;;QAtHlB,qCAuHI,qBAAc,I;;QAvHlB,qCAwHI,qBAAc,I;;QAxHlB,qCAyHI,qBAAc,I;;QAzHlB,qCA0HI,qBAAc,I;;QA1HlB,qCA2HI,qBAAc,I;;QA3HlB,qCA4HI,qBAAc,I;;QA5HlB,qCA6HI,qBAAc,I;;QA7HlB,qCA8HI,qBAAc,I;;QA9HlB,qCA+HI,qBAAc,I;;QA/HlB,qCAgII,qBAAc,I;;QAhIlB,qCAiII,qBAAc,I;;QAjIlB,qCAkII,qBAAc,I;;QAlIlB,qCAmII,qBAAc,I;;QAnIlB,qCAoII,qBAAc,I;;QApIlB,qCAqII,qBAAc,I;;QArIlB,qCAsII,qBAAc,I;;QAtIlB,qCAuII,qBAAc,I;;QAvIlB,qCAwII,qBAAc,I;;QAxIlB,qCAyII,qBAAc,I;;QAzIlB,qCA0II,qBAAc,I;;QA1IlB,qCA2II,qBAAc,I;;QA3IlB,qCA4II,qBAAc,I;;QA5IlB,qCA6II,qBAAc,I;;QA7IlB,qCA8II,qBAAc,I;;QA9IlB,qCA+II,qBAAc,I;;QA/IlB,qCAgJI,qBAAc,I;;QAhJlB,qCAiJI,qBAAc,I;;QAjJlB,qCAkJI,qBAAc,I;;QAlJlB,qCAmJI,qBAAc,I;;QAnJlB,qCAoJI,qBAAc,I;;QApJlB,qCAqJI,qBAAc,I;;QArJlB,qCAsJI,qBAAc,I;;QAtJlB,qCAuJI,qBAAc,I;;QAvJlB,qCAwJI,qBAAc,I;;QAxJlB,qCAyJI,qBAAc,I;;QAzJlB,qCA0JI,qBAAc,I;;QA1JlB,qCA2JI,qBAAc,I;;QA3JlB,qCA4JI,qBAAc,I;;QA5JlB,qCA6JI,qBAAc,I;;QA7JlB,qCA8JI,qBAAc,I;;QA9JlB,qCA+JI,qBAAc,I;;QA/JlB,qCAgKI,qBAAc,I;;QAhKlB,qCAiKI,qBAAc,I;;QAjKlB,qCAkKI,qBAAc,I;;QAlKlB,qCAmKI,qBAAc,I;;QAnKlB,qCAoKI,qBAAc,I;;QApKlB,qCAqKI,qBAAc,I;;QArKlB,qCAsKI,qBAAc,I;;QAtKlB,qCAuKI,qBAAc,I;;QAvKlB,qCAwKI,qBAAc,I;;QAxKlB,qCAyKI,qBAAc,I;;QAzKlB,qCA0KI,qBAAc,I;;QA1KlB,qCA2KI,qBAAc,I;;QA3KlB,qCA4KI,qBAAc,I;;QA5KlB,qCA6KI,qBAAc,I;;QA7KlB,qCA8KI,qBAAc,I;;QA9KlB,qCA+KI,qBAAc,I;;QA/KlB,qCAgLI,qBAAc,I;;QAhLlB,qCAiLI,qBAAc,I;;QAjLlB,qCAkLI,qBAAc,I;;QAlLlB,qCAmLI,qBAAc,I;;QAnLlB,qCAoLI,qBAAc,I;;QApLlB,qCAqLI,qBAAc,I;;QArLlB,qCAsLI,qBAAc,I;;QAtLlB,qCAuLI,qBAAc,I;;QAvLlB,qCAwLI,qBAAc,I;;QAxLlB,qCAyLI,qBAAc,I;;QAzLlB,qCA0LI,qBAAc,I;;QA1LlB,qCA2LI,qBAAc,I;;QA3LlB,qCA4LI,qBAAc,I;;QA5LlB,qCA6LI,qBAAc,I;;QA7LlB,qCA8LI,qBAAc,I;;QA9LlB,qCA+LI,qBAAc,I;;QA/LlB,qCAgMI,qBAAc,I;;QAhMlB,qCAiMI,qBAAc,I;;QAjMlB,qCAkMI,qBAAc,I;;QAlMlB,qCAmMI,qBAAc,I;;QAnMlB,qCAoMI,qBAAc,I;;QApMlB,qCAqMI,qBAAc,I;;QArMlB,qCAsMI,qBAAc,I;;QAtMlB,qCAuMI,qBAAc,I;;QAvMlB,qCAwMI,qBAAc,I;;QAxMlB,qCAyMI,qBAAc,I;;QAzMlB,qCA0MI,qBAAc,I;;QA1MlB,qCA2MI,qBAAc,I;;QA3MlB,qCA4MI,qBAAc,I;;QA5MlB,qCA6MI,qBAAc,I;;QA7MlB,qCA8MI,qBAAc,I;;QA9MlB,qCA+MI,qBAAc,I;;QA/MlB,qCAgNI,qBAAc,I;;QAhNlB,qCAiNI,qBAAc,I;;QAjNlB,qCAkNI,qBAAc,I;;QAlNlB,qCAmNI,qBAAc,I;;QAnNlB,qCAoNI,qBAAc,I;;QApNlB,qCAqNI,qBAAc,I;;QArNlB,qCAsNI,qBAAc,I;;QAtNlB,qCAuNI,qBAAc,I;;QAvNlB,qCAwNI,qBAAc,I;;QAxNlB,qCAyNI,qBAAc,I;;QAzNlB,qCA0NI,qBAAc,I;;QA1NlB,qCA2NI,qBAAc,I;;QA3NlB,qCA4NI,qBAAc,I;;QA5NlB,qCA6NI,qBAAc,I;;QA7NlB,qCA8NI,qBAAc,I;;QA9NlB,qCA+NI,qBAAc,I;;QA/NlB,qCAgOI,qBAAc,I;;QAhOlB,qCAiOI,qBAAc,I;;QAjOlB,qCAkOI,qBAAc,I;;QAlOlB,qCAmOI,qBAAc,I;;QAnOlB,qCAoOI,qBAAc,I;;QApOlB,qCAqOI,qBAAc,I;;QArOlB,qCAsOI,qBAAc,I;;QAtOlB,qCAuOI,qBAAc,I;;QAvOlB,qCAwOI,qBAAc,I;;QAxOlB,qCAyOI,qBAAc,I;;QAzOlB,qCA0OI,qBAAc,I;;QA1OlB,qCA2OI,qBAAc,I;;QA3OlB,qCA4OI,qBAAc,I;;QA5OlB,qCA6OI,qBAAc,I;;QA7OlB,qCA8OI,qBAAc,I;;QA9OlB,qCA+OI,qBAAc,I;;QA/OlB,qCAgPI,qBAAc,I;;QAhPlB,qCAiPI,qBAAc,I;;QAjPlB,qCAkPI,qBAAc,I;;QAlPlB,qCAmPI,qBAAc,I;;QAnPlB,qCAoPI,qBAAc,I;;QApPlB,qCAqPI,qBAAc,I;;QArPlB,qCAsPI,qBAAc,I;;QAtPlB,qCAuPI,qBAAc,I;;QAvPlB,qCAwPI,qBAAc,I;;QAxPlB,qCAyPI,qBAAc,I;;QAzPlB,qCA0PI,qBAAc,I;;QA1PlB,qCA2PI,qBAAc,I;;QA3PlB,qCA4PI,qBAAc,I;;QA5PlB,qCA6PI,qBAAc,I;;QA7PlB,qCA8PI,qBAAc,I;;QA9PlB,qCA+PI,qBAAc,I;;QA/PlB,qCAgQI,qBAAc,I;;QAhQlB,qCAiQI,qBAAc,I;;QAjQlB,qCAkQI,qBAAc,I;;QAlQlB,qCAmQI,qBAAc,I;;QAnQlB,qCAoQI,qBAAc,I;;QApQlB,qCAqQI,qBAAc,I;;QArQlB,qCAsQI,qBAAc,I;;QAtQlB,qCAuQI,qBAAc,I;;QAvQlB,qCAwQI,qBAAc,I;;QAxQlB,qCAyQI,qBAAc,I;;QAzQlB,qCA0QI,qBAAc,I;;QA1QlB,qCA2QI,qBAAc,I;;QA3QlB,qCA4QI,qBAAc,I;;QA5QlB,qCA6QI,qBAAc,I;;QA7QlB,qCA8QI,qBAAc,I;;QA9QlB,qCA+QI,qBAAc,I;;QA/QlB,qCAgRI,qBAAc,I;;QAhRlB,qCAiRI,qBAAc,I;;QAjRlB,qCAkRI,qBAAc,I;;QAlRlB,qCAmRI,qBAAc,I;;QAnRlB,qCAoRI,qBAAc,I;;QApRlB,qCAqRI,qBAAc,I;;QArRlB,qCAsRI,qBAAc,I;;QAtRlB,qCAuRI,qBAAc,I;;QAvRlB,qCAwRI,qBAAc,I;;QAxRlB,qCAyRI,qBAAc,I;;QAzRlB,qCA0RI,qBAAc,I;;QA1RlB,qCA2RI,qBAAc,I;;QA3RlB,qCA4RI,qBAAc,I;;QA5RlB,qCA6RI,qBAAc,I;;QA7RlB,qCA8RI,qBAAc,I;;QA9RlB,qCA+RI,qBAAc,I;;QA/RlB,qCAgSI,qBAAc,I;;QAhSlB,qCAiSI,qBAAc,I;;QAjSlB,qCAkSI,qBAAc,I;;QAlSlB,qCAmSI,qBAAc,I;;QAnSlB,qCAoSI,qBAAc,I;;QApSlB,qCAqSI,qBAAc,I;;QArSlB,qCAsSI,qBAAc,I;;QAtSlB,qCAuSI,qBAAc,I;;QAvSlB,qCAwSI,qBAAc,I;;QAxSlB,qCAySI,qBAAc,I;;QAzSlB,qCA0SI,qBAAc,I;;QA1SlB,qCA2SI,qBAAc,I;;QA3SlB,qCA4SI,qBAAc,I;;QA5SlB,qCA6SI,qBAAc,I;;QA7SlB,qCA8SI,qBAAc,I;;QA9SlB,qCA+SI,qBAAc,I;;QA/SlB,qCAgTI,qBAAc,I;;QAhTlB,qCAiTI,qBAAc,I;;QAjTlB,qCAkTI,qBAAc,I;;QAlTlB,qCAmTI,qBAAc,I;;QAnTlB,qCAoTI,qBAAc,I;;QApTlB,qCAqTI,qBAAc,I;;QArTlB,qCAsTI,qBAAc,I;;QAtTlB,qCAuTI,qBAAc,I;;QAvTlB,qCAwTI,qBAAc,I;;QAxTlB,qCAyTI,qBAAc,I;;QAzTlB,qCA0TI,qBAAc,I;;QA1TlB,qCA2TI,qBAAc,I;;QA3TlB,qCA4TI,qBAAc,I;;QA5TlB,qCA6TI,qBAAc,I;;QA7TlB,qCA8TI,qBAAc,I;;QA9TlB,qCA+TI,qBAAc,I;;QA/TlB,qCAgUI,qBAAc,I;;QAhUlB,qCAiUI,qBAAc,I;;QAjUlB,qCAkUI,qBAAc,I;;QAlUlB,qCAmUI,qBAAc,I;;QAnUlB,qCAoUI,qBAAc,I;;QApUlB,qCAqUI,qBAAc,I;;QArUlB,qCAsUI,qBAAc,I;;QAtUlB,qCAuUI,qBAAc,I;;QAvUlB,qCAwUI,qBAAc,I;;QAxUlB,qCAyUI,qBAAc,I;;QAzUlB,qCA0UI,qBAAc,I;;QA1UlB,qCA2UI,qBAAc,I;;QA3UlB,qCA4UI,qBAAc,I;;QA5UlB,qCA6UI,qBAAc,I;;QA7UlB,qCA8UI,qBAAc,I;;QA9UlB,qCA+UI,qBAAc,I;;QA/UlB,qCAgVI,qBAAc,I;;QAhVlB,qCAiVI,qBAAc,I;;QAjVlB,qCAkVI,qBAAc,I;;QAlVlB,qCAmVI,qBAAc,I;;QAnVlB,qCAoVI,qBAAc,I;;QApVlB,qCAqVI,qBAAc,I;;QArVlB,qCAsVI,qBAAc,I;;QAtVlB,qCAuVI,qBAAc,I;;QAvVlB,qCAwVI,qBAAc,I;;QAxVlB,qCAyVI,qBAAc,I;;QAzVlB,qCA0VI,qBAAc,I;;QA1VlB,qCA2VI,qBAAc,I;;QA3VlB,qCA4VI,qBAAc,I;;QA5VlB,qCA6VI,qBAAc,I;;QA7VlB,qCA8VI,qBAAc,I;;QA9VlB,qCA+VI,qBAAc,I;;QA/VlB,qCAgWI,qBAAc,I;;QAhWlB,qCAiWI,qBAAc,I;;QAjWlB,qCAkWI,qBAAc,I;;QAlWlB,qCAmWI,qBAAc,I;;QAnWlB,qCAoWI,qBAAc,I;;QApWlB,qCAqWI,qBAAc,I;;QArWlB,qCAsWI,qBAAc,I;;QAtWlB,qCAuWI,qBAAc,I;;QAvWlB,qCAwWI,qBAAc,I;;QAxWlB,qCAyWI,qBAAc,I;;QAzWlB,qCA0WI,qBAAc,I;;QA1WlB,qCA2WI,qBAAc,I;;QA3WlB,qCA4WI,qBAAc,I;;QA5WlB,qCA6WI,qBAAc,I;;QA7WlB,qCA8WI,qBAAc,I;;QA9WlB,qCA+WI,qBAAc,I;;QA/WlB,qCAgXI,qBAAc,I;;QAhXlB,qCAiXI,qBAAc,I;;QAjXlB,qCAkXI,qBAAc,I;;QAlXlB,qCAmXI,qBAAc,I;;QAnXlB,qCAoXI,qBAAc,I;;QApXlB,qCAqXI,qBAAc,I;;QArXlB,qCAsXI,qBAAc,I;;QAtXlB,qCAuXI,qBAAc,I;;QAvXlB,qCAwXI,qBAAc,I;;QAxXlB,qCAyXI,qBAAc,I;;QAzXlB,qCA0XI,qBAAc,I;;QA1XlB,qCA2XI,qBAAc,I;;QA3XlB,qCA4XI,qBAAc,I;;QA5XlB,qCA6XI,qBAAc,I;;QA7XlB,qCA8XI,qBAAc,I;;QA9XlB,qCA+XI,qBAAc,I;;QA/XlB,qCAgYI,qBAAc,I;;QAhYlB,qCAiYI,qBAAc,I;;QAjYlB,qCAkYI,qBAAc,I;;QAlYlB,qCAmYI,qBAAc,I;;QAnYlB,qCAoYI,qBAAc,I;;QApYlB,qCAqYI,qBAAc,I;;QArYlB,qCAsYI,qBAAc,I;;QAtYlB,qCAuYI,qBAAc,I;;QAvYlB,qCAwYI,qBAAc,I;;QAxYlB,qCAyYI,qBAAc,I;;QAzYlB,qCA0YI,qBAAc,I;;QA1YlB,qCA2YI,qBAAc,I;;QA3YlB,qCA4YI,qBAAc,I;;QA5YlB,qCA6YI,qBAAc,I;;QA7YlB,qCA8YI,qBAAc,I;;QA9YlB,qCA+YI,qBAAc,I;;QA/YlB,qCAgZI,qBAAc,I;;QAhZlB,qCAiZI,qBAAc,I;;QAjZlB,qCAkZI,qBAAc,I;;QAlZlB,qCAmZI,qBAAc,I;;QAnZlB,qCAoZI,qBAAc,I;;QApZlB,qCAqZI,qBAAc,I;;QArZlB,qCAsZI,qBAAc,I;;QAtZlB,qCAuZI,qBAAc,I;;QAvZlB,qCAwZI,qBAAc,I;;QAxZlB,qCAyZI,qBAAc,I;;QAzZlB,qCA0ZI,qBAAc,I;;QA1ZlB,qCA2ZI,qBAAc,I;;QA3ZlB,qCA4ZI,qBAAc,I;;QA5ZlB,qCA6ZI,qBAAc,I;;QA7ZlB,qCA8ZI,qBAAc,I;;QA9ZlB,qCA+ZI,qBAAc,I;;QA/ZlB,qCAgaI,qBAAc,I;;QAhalB,qCAiaI,qBAAc,I;;QAjalB,qCAkaI,qBAAc,I;;QAlalB,qCAmaI,qBAAc,I;;QAnalB,qCAoaI,qBAAc,I;;QApalB,qCAqaI,qBAAc,I;;QAraJ,qCAsaI,qBAAc,I;;QAtalB,qCAuaI,qBAAc,I;;QAvalB,qCAwaI,qBAAc,I;;QAxalB,qCAyaI,qBAAc,I;;QAzalB,qCA0aI,qBAAc,I;;QA1alB,qCA2aI,qBAAc,I;;QA3alB,qCA4aI,qBAAc,I;;QA5alB,qCA6aI,qBAAc,I;;QA7alB,qCA8aI,qBAAc,I;;QA9alB,qCA+aI,qBAAc,I;;QA/alB,qCAgbI,qBAAc,I;;QAhblB,qCAibI,qBAAc,I;;QAjblB,qCAkbI,qBAAc,I;;QAlblB,qCAmbI,qBAAc,I;;QAnblB,qCAobI,qBAAc,I;;QApblB,qCAqbI,qBAAc,I;;QArblB,qCAsbI,qBAAc,I;;QAtblB,qCAubI,qBAAc,I;;QAvblB,qCAwbI,qBAAc,I;;QAxblB,qCAybI,qBAAc,I;;QAzblB,qCA0bI,qBAAc,I;;QA1blB,qCA2bI,qBAAc,I;;QA3blB,qCA4bI,qBAAc,I;;QA5blB,qCA6bI,qBAAc,I;;QA7blB,qCA8bI,qBAAc,I;;QA9blB,qCA+bI,qBAAc,I;;QA/blB,qCAgcI,qBAAc,I;;QAhclB,qCAicI,qBAAc,I;;QAjclB,qCAkcI,qBAAc,I;;QAlclB,qCAmcI,qBAAc,I;;QAncJ,qCAocI,qBAAc,I;;QApclB,qCAqcI,qBAAc,I;;QArclB,qCAscI,qBAAc,I;;QAtclB,qCAucI,qBAAc,I;;QAvclB,qCAwcI,qBAAc,I;;QAxclB,qCAycI,qBAAc,I;;QAzclB,qCA0cI,qBAAc,I;;QA1clB,qCA2cI,qBAAc,I;;QA3clB,qCA4cI,qBAAc,I;;QA5clB,qCA6cI,qBAAc,I;;QA7clB,qCA8cI,qBAAc,I;;QA9clB,qCA+cI,qBAAc,I;;QA/clB,qCAgdI,qBAAc,I;;QAhdlB,qCAidI,qBAAc,I;;QAjdlB,qCAkdI,qBAAc,I;;QAldlB,qCAmdI,qBAAc,I;;QAndlB,qCAodI,qBAAc,I;;QApdlB,qCAqdI,qBAAc,I;;QArdlB,qCAsdI,qBAAc,I;;QAtdlB,qCAudI,qBAAc,I;;QAvdlB,qCAwdI,qBAAc,I;;QAxdlB,qCAydI,qBAAc,I;;QAzdlB,qCA0dI,qBAAc,I;;QA1dlB,qCA2dI,qBAAc,I;;QA3dlB,qCA4dI,qBAAc,I;;QA5dlB,qCA6dI,qBAAc,I;;QA7dlB,qCA8dI,qBAAc,I;;QA9dlB,qCA+dI,qBAAc,I;;QA/dlB,qCAgeI,qBAAc,I;;QAheJ,qCAieI,qBAAc,I;;QAjelB,qCAkeI,qBAAc,I;;QAlelB,qCAmeI,qBAAc,I;;QAneJ,qCAoeI,qBAAc,I;;QApelB,qCAqeI,qBAAc,I;;QAreJ,qCAseI,qBAAc,I;;QAteJ,qCAueI,qBAAc,I;;QAveJ,qCAweI,qBAAc,I;;QAxeJ,qCAyeI,qBAAc,I;;QAzeJ,qCA0eI,qBAAc,I;;QA1eJ,qCA2eI,qBAAc,I;;QA3eJ,qCA4eI,qBAAc,I;;QA5eJ,qCA6eI,qBAAc,I;;QA7eJ,qCA8eI,qBAAc,I;;QA9eJ,qCA+eI,qBAAc,I;;QA/eJ,qCAgfI,qBAAc,I;;QAhfJ,qCAifI,qBAAc,I;;QAjfJ,qCAkfI,qBAAc,I;;QAlfJ,qCAmfI,qBAAc,I;;QAnfJ,qCAofI,qBAAc,I;;QApfJ,qCAqfI,qBAAc,I;;QArfJ,qCAsfI,qBAAc,I;;QAtfJ,qCAufI,qBAAc,I;;QAvfJ,qCAwfI,qBAAc,I;;QAxfJ,qCAyfI,qBAAc,I;;QAzfJ,qCA0fI,qBAAc,I;;QA1fJ,qCA2fI,qBAAc,I;;QA3fJ,qCA4fI,qBAAc,I;;QA5fJ,qCA6fI,qBAAc,I;;QA7fJ,qCA8fI,qBAAc,I;;QA9fJ,qCA+fI,qBAAc,I;;QA/fJ,qCAggBI,qBAAc,I;;QAhgBJ,qCAigBI,qBAAc,I;;QAjgBJ,qCAkgBI,qBAAc,I;;QAlgBJ,qCAmgBI,qBAAc,I;;QAngBJ,qCAogBI,qBAAc,I;;QApgBJ,qCAqgBI,qBAAc,I;;QArgBJ,qCAsgBI,qBAAc,I;;QAtgBJ,qCAugBI,qBAAc,I;;QAvgBJ,qCAwgBI,qBAAc,I;;QAxgBJ,qCAygBI,qBAAc,I;;QAzgBJ,qCA0gBI,qBAAc,I;;QA1gBJ,qCA2gBI,qBAAc,I;;QA3gBJ,qCA4gBI,qBAAc,I;;QA5gBJ,qCA6gBI,qBAAc,I;;QA7gBJ,qCA8gBI,qBAAc,I;;QA9gBJ,qCA+gBI,qBAAc,I;;QA/gBJ,qCAghBI,qBAAc,I;;QAhhBJ,qCAihBI,qBAAc,I;;QAjhBJ,qCAkhBI,qBAAc,I;;QAlhBJ,qCAmhBI,qBAAc,I;;QAnhBJ,qCAohBI,qBAAc,I;;QAphBJ,qCAqhBI,qBAAc,I;;QArhBJ,qCAshBI,qBAAc,I;;QAthBJ,qCAuhBI,qBAAc,I;;QAvhBJ,qCAwhBI,qBAAc,I;;QAxhBJ,qCAyhBI,qBAAc,I;;QAzhBJ,qCA0hBI,qBAAc,I;;QA1hBJ,qCA2hBI,qBAAc,I;;QA3hBJ,qCA4hBI,qBAAc,I;;QA5hBJ,qCA6hBI,qBAAc,I;;QA7hBJ,qCA8hBI,qBAAc,I;;QA9hBJ,qCA+hBI,qBAAc,I;;QA/hBJ,qCAgiBI,qBAAc,I;;QAhiBJ,qCAiiBI,qBAAc,I;;QAjiBJ,qCAkiBI,qBAAc,I;;QAliBJ,qCAmiBI,qBAAc,I;;QAniBJ,qCAoiBI,qBAAc,I;;QApiBJ,qCAqiBI,qBAAc,I;;QAriBJ,qCAsiBI,qBAAc,I;;QAtiBJ,qCAuiBI,qBAAc,I;;QAviBJ,qCAwiBI,qBAAc,I;;QAxiBJ,qCAyiBI,qBAAc,I;;QAziBJ,qCA0iBI,qBAAc,I;;QA1iBJ,qCA2iBI,qBAAc,I;;QA3iBJ,qCA4iBI,qBAAc,I;;QA5iBJ,qCA6iBI,qBAAc,I;;QA7iBJ,qCA8iBI,qBAAc,I;;QA9iBJ,qCA+iBI,qBAAc,I;;QA/iBJ,qCAgjBI,qBAAc,I;;QAhjBJ,qCAijBI,qBAAc,I;;QAjjBJ,qCAkjBI,qBAAc,I;;QAljBJ,qCAmjBI,qBAAc,I;;QAnjBJ,qCAojBI,qBAAc,I;;QApjBJ,qCAqjBI,qBAAc,I;;QArjBJ,qCAsjBI,qBAAc,I;;QAtjBJ,qCAujBI,qBAAc,I;;QAvjBJ,qCAwjBI,qBAAc,I;;QAxjBJ,qCAyjBI,qBAAc,I;;QAzjBJ,qCA0jBI,qBAAc,I;;QA1jBJ,qCA2jBI,qBAAc,I;;QA3jBJ,qCA4jBI,qBAAc,I;;QA5jBJ,qCA6jBI,qBAAc,I;;QA7jBJ,qCA8jBI,qBAAc,I;;QA9jBJ,qCA+jBI,qBAAc,I;;QA/jBJ,qCAgkBI,qBAAc,I;;QAhkBJ,qCAikBI,qBAAc,I;;QAjkBJ,qCAkkBI,qBAAc,I;;QAlkBJ,qCAmkBI,qBAAc,I;;QAnkBJ,qCAokBI,qBAAc,I;;QApkBJ,qCAqkBI,qBAAc,I;;QArkBJ,qCAskBI,qBAAc,I;;QAtkBJ,qCAukBI,qBAAc,I;;QAvkBJ,qCAwkBI,qBAAc,I;;QAxkBJ,qCAykBI,qBAAc,I;;QAzkBJ,qCA0kBI,qBAAc,I;;QA1kBJ,qCA2kBI,qBAAc,I;;QA3kBJ,qCA4kBI,qBAAc,I;;QA5kBJ,qCA6kBI,qBAAc,I;;QA7kBJ,qCA8kBI,qBAAc,I;;QA9kBJ,qCA+kBI,qBAAc,I;;QA/kBJ,qCAglBI,qBAAc,I;;QAhlBJ,qCAilBI,qBAAc,I;;QAjlBJ,qCAklBI,qBAAc,I;;QAllBJ,qCAmlBI,qBAAc,I;;QAnlBJ,qCAolBI,qBAAc,I;;QAplBJ,qCAqlBI,qBAAc,I;;QArlBJ,qCAslBI,qBAAc,I;;QAtlBJ,qCAulBI,qBAAc,I;;QAvlBJ,qCAwlBI,qBAAc,I;;QAxlBJ,qCAylBI,qBAAc,I;;QAzlBJ,qCA0lBI,qBAAc,I;;QA1lBJ,qCA2lBI,qBAAc,I;;QA3lBJ,qCA4lBI,qBAAc,I;;QA5lBJ,qCA6lBI,qBAAc,I;;QA7lBJ,qCA8lBI,qBAAc,I;;QA9lBJ,qCA+lBI,qBAAc,I;;QA/lBJ,qCAgmBI,qBAAc,I;;QAhmBJ,qCAimBI,qBAAc,I;;QAjmBJ,qCAkmBI,qBAAc,I;;QAlmBJ,qCAmmBI,qBAAc,I;;QAnmBJ,qCAomBI,qBAAc,I;;QApmBJ,qCAqmBI,qBAAc,I;;QArmBJ,qCAsmBI,qBAAc,I;;QAtmBJ,qCAumBI,qBAAc,I;;QAvmBJ,qCAwmBI,qBAAc,I;;QAxmBJ,qCAymBI,qBAAc,I;;QAzmBJ,qCA0mBI,qBAAc,I;;QA1mBJ,qCA2mBI,qBAAc,I;;QA3mBJ,qCA4mBI,qBAAc,I;;QA5mBJ,qCA6mBI,qBAAc,I;;QA7mBJ,qCA8mBI,qBAAc,I;;QA9mBJ,qCA+mBI,qBAAc,I;;QA/mBJ,qCAgnBI,qBAAc,I;;QAhnBJ,qCAinBI,qBAAc,I;;QAjnBJ,qCAknBI,qBAAc,I;;QAlnBJ,qCAmnBI,qBAAc,I;;QAnnBJ,qCAonBI,qBAAc,I;;QApnBJ,qCAqnBI,qBAAc,I;;QArnBJ,qCAsnBI,qBAAc,I;;QAtnBJ,qCAunBI,qBAAc,I;;QAvnBJ,qCAwnBI,qBAAc,I;;QAxnBJ,qCAynBI,qBAAc,I;;QAznBJ,qCA0nBI,qBAAc,I;;QA1nBJ,qCA2nBI,qBAAc,I;;QA3nBJ,qCA4nBI,qBAAc,I;;QA5nBJ,qCA6nBI,qBAAc,I;;QA7nBJ,qCA8nBI,qBAAc,I;;QA9nBJ,qCA+nBI,qBAAc,I;;QA/nBJ,qCAgoBI,qBAAc,I;;QAhoBJ,qCAioBI,qBAAc,I;;QAjoBJ,qCAkoBI,qBAAc,I;;QAloBJ,qCAmoBI,qBAAc,I;;QAnoBJ,qCAooBI,qBAAc,I;;QApoBJ,qCAqoBI,qBAAc,I;;QAroBJ,qCAsoBI,qBAAc,I;;QAtoBJ,qCAuoBI,qBAAc,I;;QAvoBJ,qCAwoBI,qBAAc,I;;QAxoBJ,qCAyoBI,qBAAc,I;;QAzoBJ,qCA0oBI,qBAAc,I;;QA1oBJ,qCA2oBI,qBAAc,I;;QA3oBJ,qCA4oBI,qBAAc,I;;QA5oBJ,qCA6oBI,qBAAc,I;;QA7oBJ,qCA8oBI,qBAAc,I;;QA9oBJ,qCA+oBI,qBAAc,I;;QA/oBJ,qCAgpBI,qBAAc,I;;QAh

A,kC;MAOI,OAAO,mBAAQ,SAAR,C;K;IAEX,mC;MAQI,IAAI,8CAAJ,C;QAA6B,MAAM,eAAM,U;K;gFAG7C,yB;MAAA,4B;MAAA,qB;MAxCQ,kD;MAwCR,wB;QAOW,Q;;UACI,OAlDH,WAkDW,OAlDX,C;;UAmDN,gC;YACS,OA3CH,WAAO,cA2CI,CA3CJ,CAAP,C;;YAwCD,O;;QAAP,W;O;KAPJ,C;kFAcA,yB;MAAA,4B;MAAA,qB;MAtDQ,kD;MAsDR,mC;QAOW,Q;;UACI,OAhEH,WAgEW,gBAhEX,C;;UAiEN,gC;YACS,OAzDH,WAAO,cAyDI,CAzDJ,CAAP,C;;YAsDD,O;;QAAP,W;O;KAPJ,C;8EAgBA,yB;MAAA,oD;MAAA,gB;MAAA,8B;MAAA,4B;QAUW,Q;QADP,yB;QACA,OAAO,gF;O;KAVX,C;+EAaA,yB;MAAA,gB;MAAA,8B;MAAA,uC;QAegB,UADL,M;QAAM,gBAAgB,2B;QACzB,sB;UAAQ,yF;;UACA,mBAAU,SAAV,C;QAFZ,a;O;KAdJ,C;kFAoBA,yB;MAAA,gB;MAAA,8B;MAAA,0C;QAUW,Q;QADP,IAAI,mBAAJ,C;UAAe,OAAO,Y;QACtB,OAAO,gF;O;KAVX,C;qEAaA,yB;MAAA,gB;MAAA,8B;MAAA,kD;QAiB0B,0B,UADf,M;QAAM,gBAAgB,2B;QACrB,sB;UAAQ,mBAAU,gBAAV,C;;UAAU,mBAAU,SAAV,C;QAFpB,a;O;KAhBJ,C;mEAsBA,yB;MAAA,4B;MAAA,gB;MAAA,8B;MAAA,uC;QAAA,a;QAAA,O;KAAA,C;mEAA

CjBkP7B,eAAW,oBAAL,SAAK,CAAL,UAAN,CiBlP6B,MAAK,KjBAI,KiBAO,KAAX,CAAhB,C;O;KjBFvD,C;8FAIA,yB;MA6NA,6B;MA7NA,wB;QAEuD,OCMD,cAAU,CD8N7B,cAAU,SAAL,GAAiB,GAAtB,CC9N6B,MAAK,GAAY,CD8N9C,cApOwC,KAoO9B,KAAL,GAAiB,GAAtB,CC9N8C,MAAZ,IAAf,C;O;KDRtD,C;8FAGA,yB;MA0NA,6B;MA1NA,wB;QAEwD,OCGF,cAAU,CD8N7B,cAAU,SAAL,GAAiB,GAAtB,CC9N6B,MAAK,GAAY,CC+N9C,cFlOyC,KEkO/B,KAAL,GAAiB,GAAtB,CD/N8C,MAAZ,IAAf,C;O;KDLtD,C;8FAGA,yB;MAuNA,6B;MAvNA,wB;QAEsD,OCAA,cAAU,CD8N7B,cAAU,SAAL,GAAiB,GAAtB,CC9N6B,MAAK,GAAY,GDAK,KCAO,KAAZ,IAAf,C;O;KDFtD,C;8FAGA,yB;MA8NA,WAAS6D,wB;MAT7D,+B;MA9NA,wB;QAEwD,OiBAA,eAAAW,CjBqO9B,eAAA,WAAoB,CjBrO9B,oBAAA,MAAK,UAAL,GAAK,KAAL,GAAK,KAAL,GAAK,K

MAAA,+B;M6LpOJ,0B;M7LoOI,wB;QAEmD,sB6LnOgC,O7LmO1B,IAAK,K6LnOX,G7LmOoB,KAAM,K6LnOM,C7LmOhC,C;O;KAFnD,C;0EAGA,yB;MAAA,+B;M6LlOJ,0B;M7LkOI,mB;QAEiC,sB6LjOqB,OAAP,C7LiOR,S6LjOe,C7LiOrB,C;O;KAFjC,C;gFAIA,Y;MASmC,gB;K;kFACnC,yB;M6L1OJ,4B;M7L0OI,mB;QASqC,O6LhPiD,Q7LgP5C,S6LhPY,G7LgPE,G6LhP8B,C;O;K7LuOtF,C;8EAUA,Y;MASiC,OAAK,SAAL,GAAiB,G;K;gFAClD,yB;MAAA,WASqD,wB;MATrD,mB;QASmC,OAAK,oBAAL,SAAK,CAAL,U;O;KATnC,C;kFAWA,Y;MAEqC,W;K;oFACrC,yB;MAAA,iC;M6L5QJ,4B;M7L4QI,mB;QASuC,uB6LlR+C,Q7LkRnC,S6LlRG,G7LkRW,G6LlRqB,C7LkR/C,C;O;KATvC,C;gFAUA,yB;MAAA,6B;MAAA,mB;QASmC,qBAAU,SAAL,GAAiB,GAAtB,C;O;KATnC,C;kFAUA,yB;MAAA,WAS6D,wB;MAT7D,+B;MAAA,mB;QASqC,sBAAW,oBAAL,SAAK,CAAL,UAAN,C;O;KATrC,C;kFAWA,Y;MAMqC,OApDC,SAAL,GAAiB,G;K;oFAqDlD,Y;MAMuC,OA3DD,SAAL,GAAiB,G;K;+BA6DlD,Y;MAAyC,OAAQ,CA7DX,SAAL,GAAiB,GA6DD,Y;K;;;;;+BA1UrD,Y;MAAA,c;MAG4D,qD;MAH5D,a;K;6BAAA,iB;MAAA,2IAG4D,oCAH5D,G;K;wEA8UA,yB;MAAA,+B;MAAA,4B;QAU0C,sBAAW,SAAN,C,C;O;KAV/C,C;0EAWA,yB;MAAA,0B;MAAA,+B;MAAA,4B;QAW2C,sBAAW,OAAL,SAAK,CAAL,UAAX,C,C;O;KAXhD,C;0EAYA,yB;MAAA,0B;MAAA,+B;MAAA,4B;QAWyC,sBAAW,OAAL,SAAK,CAAL,UAAX,C,C;O;KAX9C,C;0EAYA,yB;MAAA,0B;MAAA,+B;MAAA,4B;QAW0C,sBAAW,OAAL,SAAK,CAAL,UAAX,C,C;O;KAX/

K,EAAY,CDiN9C,cCvN+B,KDuNrB,KAAL,GAAiB,GAAtB,CCjN8C,MAAZ,CAAf,C;O;KARtD,C;4FAGA,yB;MC8MA,6B;MD9MA,wB;QAEwD,OAGF,cAAe,YAAV,IAAK,KAAK,EAAY,CCkN9C,cDrNgC,KCqNtB,KAAL,GAAiB,KAAtB,CDlN8C,MAAZ,CAAf,C;O;KALtD,C;4FAGA,yB;MAAA,6B;MAAA,wB;QAEsD,qBAAe,YAAV,IAAK,KAAK,EAAM,KAAM,KAAZ,EAAM,C;O;KAFtD,C;4FAGA,yB;MAqOA,kBAS6D,sB;MAT7D,+B;MArOA,wB;QAEwD,OgBAA,eAAW,ChB4O9B,eAAW,oBAAL,SAAK,CAAL,iBAAN,CgB5O8C,8B,MAAK,UhBAK,KgBAO,KAAZ,EAAM,C;O;KAhBtD,C;wFAGA,yB;MDmMA,6B;MC7LA,4C;MANA,wB;QAEqD,qBAAe,YAAV,IAAK,KAAK,EAAM,KAAM,KAAZ,EAAM,C;O;KARtD,C;4FAGA,yB;MCiMA,6B;MD9LA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,ECqMjB,cDxM4B,KCwMlB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwNA,kBAS6D,sB;MAT7D,+B;MgBxNA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgBwMjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MDsLA,6B;MC7KA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MCiLA,6B;MD9KA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,ECqLjB,cDxL4B,KCwLlB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwMA,kBAS6D,sB;MAT7D,+B;MgBvMA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgBuLjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MDqKA,6B;MC7JA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MCiKA,6B;MD9JA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,ECqKjB,cDxK4B,KCwKlB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwLA,kBAS6D,sB;MAT7D,+B;MgBtLA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgBsKjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MDoJA,6B;MC7IA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MCgJA,6B;MD9IA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,ECoJjB,cDxJ4B,KCwJlB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwKA,kBAS6D,sB;MAT7D,+B;MgBrKA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgBqJjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MDmIA,6B;MC7HA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MC+IA,6B;MD9HA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,ECmIjB,cDxI4B,KCwIlB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwJA,kBAS6D,sB;MAT7D,+B;MgBpJA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgBoIjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MDkHA,6B;MC7GA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MC8HA,6B;MD9GA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,ECkHjB,cDxH4B,KCwHlB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwIA,kBAS6D,sB;MAT7D,+B;MgBnIA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgBmHjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MDiGA,6B;MC7FA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MC6GA,6B;MD9FA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,ECiGjB,cDxG4B,KCwGlB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwHA,kBAS6D,sB;MAT7D,+B;MgBlHA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgBkGjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MDgFA,6B;MC7EA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MC4FA,6B;MD9EA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,ECgFjB,cDxF4B,KCwFlB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwGA,kBAS6D,sB;MAT7D,+B;MgBjGA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgBiFjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MD+DA,6B;MC7DA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MC2EA,6B;MD9DA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,ECgEjB,cDxE4B,KCwElB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwFA,kBAS6D,sB;MAT7D,+B;MgBhFA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgBgEjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MD8CA,6B;MC7CA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MC0DA,6B;MD9CA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EC+CjB,cDxD4B,KCwDlB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwEA,kBAS6D,sB;MAT7D,+B;MgB/EA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgB+CjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MD6BA,6B;MC7BA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MCyCA,6B;MD9BA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EC8BjB,cDxC4B,KCwClB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwDA,kBAS6D,sB;MAT7D,+B;MgB9DA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgB8BjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MD4AA,6B;MC7AA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MCwBA,6B;MD9AA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EC6AjB,cDxB4B,KCwBlB,lB,KAAK,GAAL,iBAAN,CAAM,C;O;KALpD,C;wFAGA,yB;MAAA,4C;MAAA,wB;QAEoD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFpD,C;wFAGA,yB;MAwCA,kBAS6D,sB;MAT7D,+B;MgB7CA,8C;MhBAA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EAAX,iBAAN,CgB6AjB,cAAW,oBAAL,SAAK,CAAL,iBAAN,CAAM,C;O;KAhBpD,C;wFAGA,yB;MD2BA,6B;MC7BA,4C;MAoCA,wB;QAEqD,kBAAW,IAAX,EAAW,EAAX,iBAAN,CAAM,C;O;KAFrD,C;wFAGA,yB;MCuAA,6B;MD9BA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EAAW,EC4BjB,cDxC4B,KCwCl

;K;sCAKjB,iB;MAM4C,OhCuXT,SgCvXS,aAAQ,KAAR,ChCuXT,C;K;sCgCrXnC,wB;MAOI,aAAQ,KAAR,IAA
iB,KhCyQY,K;K;iFgCrQH,Y;MAAQ,OAAA,YAAQ,O;K;mCAE9C,Y;MAC6E,8BAAS,YAAT,C;K;IAGvD,mC;
MAAgC,uB;MAA/B,oB;MACnB,eAAoB,C;K;2CACpB,Y;MAAyB,sBAAQ,YAAM,O;K;4CACvC,Y;MAAwD,Q;
MAA9B,IAAI,eAAQ,YAAM,OAAlB,C;QAAA,OhCiWK,SgCjWmB,aAAM,mBAAN,EAAM,2BAAN,OhCiWnB,
C;;QgCjWgD,MAAM,2BAAuB,YAAM,WAA7B,C;K;;yCAGzF,mB;MAIS,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,
C;QAAgC,OAAO,K;MAEvC,OAAe,WAAR,YAAQ,EAAS,OhCmPK,KgCnPd,C;K;8CAGnB,oB;MACY,Q;MAA
2B,gBAA3B,gE;MA2B,c;;Qf0nDvB,U;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,aAAO,I;UAAP,e;;QACrB,
6B;QAAhB,OAAgB,gBAAhB,C;UAAgB,2B;Ue1nD6B,2Bf0nDR,Oe1nDQ,O;UAAA,W;YAAsB,oBAAR,YAAQ,
Ef0nD9B,OjB34CJ,KgC/OkC,C;;Uf0nD7C,IAAI,OAAJ,C;YAAyB,aAAO,K;YAAP,e;;;QAC/C,C,aAAO,I;;;Me3nDH
,iB;K;kCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,KAAqB,C;K;;IA/CvD,qC;MAAA,mD;MACgC,sBAAK,eA
AS,IAAT,CAAL,C;MADhC,Y;K;;;;mCAPJ,Y;MAAA,OAKqB,oDALrB,M;K;mCAAA,Y;MAAA,c;MAKqB,wD;
MALrB,a;K;iCAAA,iB;MAAA,2IAKqB,0CALrB,G;K;8EAyDA,yB;MAAA,uC;MAWoC,wC;QAAA,wB;UAAW,
OAAA,aAAK,KAAL,ChC8NV,K;S;O;MgCzOrC,6B;QAWI,OAAO,mBAAU,gCAAS,IAAT,GAAe,sBAAf,CAAV
,C;O;KAXX,C;gFAcA,oB;MAGkE,e;K;I6LnE5C,wC;MAsBlB,iC;MAtBsD,2BAAgB,KAAhB,EAAuB,YAAvB,E
AAqC,CAArC,C;K;kFAC7B,Y;MAAQ,iB;K;yFACD,Y;MAAQ,gB;K;2CAExC,iB;MAA8C,W7NwCoB,Y6NxCp
B,U7NwCqC,KAAjB,E6NxCX,K7NwCwC,KAA7B,C6NxCpB,K;MAAA,S;QAAkB,O7NwCE,Y6NxCF,K7NwC
mB,KAAjB,E6NxCO,S7NwCsB,KAA7B,C6NxCF,K;;MAAlB,W;K;kCAE9C,Y;MAKkC,O7NiCgC,Y6NjChC,U7
NiCiD,KAAjB,E6NjCxB,S7NiCqD,KAA7B,C6NjChC,I;K;iCAElC,iB;MAEY,UAAwB,M;MADhC,2CAAuB,kBA
Aa,KAAM,UAAnB,KACf,2CAAS,KAAM,MAAf,cAAwB,6CAAQ,KAAM,KAAd,QAAxB,CADe,CAAvB,C;K;m
CAGJ,Y;MACI,OAAI,cAAJ,GAAe,EAAf,GAAwB,MAAK,U7NyQA,K6NzQL,QAAqB,S7NyQhB,K6NzQL,I;K;
mCAE5B,Y;MAAkC,OAAE,UAAF,qBAAU,S;K;IAE5C,+B;MAAA,mC;MACI,aAC8B,cAAU,4BAAK,UAAf,EA
A0B,4BAAK,UAA/B,C;K;;;IAFlC,2C;MAAA,0C;QAAA,yB;;MAAA,mC;K;;IAYJ,oD;MA4CI,uC;MAtCI,IAAI,S
AAQ,CAAZ,C;QAAuB,MAAa,gCAAyB,wBAAzB,C;MACpC,IAAI,SAAQ,WAAZ,C;QAA2B,MAAa,gCAAyB,w
EAAzB,C;MAG5C,aAGyB,K;MAEzB,YAGwB,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAExB,YAGuB,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAEvB,YAGsB,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAEtB,YAGqB,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAErB,YAGoB,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAEpB,YAGmB,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAEnB,YAGkB,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAElB,YAGiB,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAEjB,YAGgB,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAEhB,YAGe,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAEf,YAGc,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAEd,YAGa,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAEb,YAGY,4BAAc,KAAd,EAAqB,YAArB,C,C;MAEZ,YAGW,4BAAa,KAAb,EAAoB,YAApB,C,C;MAEX,YAGU,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAEV,YAGS,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAET,YAGQ,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAER,YAGO,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEP,YAGM,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEN,YAGK,4BAAO,KAAP,EAAc,YAAd,C,C;MAEL,YAGI,4BAAM,KAAN,EAAa,YAAb,C,C;MAEJ,YAGG,4BAAK,KAAL,EAAY,YAAZ,C,C;MAEH,YAGE,4BAAI,KAAJ,EAAW,YAAX,C,C;MAEF,YAGC,4BAAG,KAAH,EAAU,YAAV,C,C;MAED,YAGA,4BAAE,KAAF,EAAS,YAAT,C,C;MAEA,YAGD,4BAAC,KAAD,EAAQ,YAAR,C,C;MAEC,YAGF,4BAAA,KAAA,EAAO,YAAP,C,C;MAEE,YAGH,4BAAC,KAAD,EAAQ,YAAR,C,C;MAEG,YAGJ,4BAAE,KAAF,EAAS,YAAT,C,C;MAEI,YAGL,4BAAG,KAAH,EAAU,YAAV,C,C;MAEK,YAGN,4BAAI,KAAJ,EAAW,YAAX,C,C;MAEM,YAGP,4BAAK,KAAL,EAAY,YAAZ,C,C;MAEO,YAGR,4BAAM,KAAN,EAAa,YAAb,C,C;MAEQ,YAGT,4BAAO,KAAP,EAAc,YAAd,C,C;MAES,YAGV,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEU,YAGX,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEW,YAGZ,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAEY,YAGb,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAEa,YAGd,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAEc,YAGf,4BAAa,KAAb,EAAoB,YAApB,C,C;MAEe,YAGhB,4BAAc,KAAd,EAAqB,YAArB,C,C;MAEgB,YAGjB,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAEiB,YAGlB,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAEkB,YAGnB,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAEmB,YAGpB,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAEoB,YAGrB,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAEqB,YAGtB,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAEsB,YAGvB,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAEuB,YAGxB,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAEwB,YAGzB,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAEyB,YAG1B,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAE0B,YAG3B,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAE2B,YAG5B,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAE4B,YAG7B,4BAAA,KAAA,EAAO,YAAP,C,C;MAE6B,YAG9B,4BAAC,KAAD,EAAQ,YAAR,C,C;MAE8B,YAG/B,4BAAE,KAAF,EAAS,YAAT,C,C;MAE+B,YAGhC,4BAAG,KAAH,EAAU,YAAV,C,C;MAEgC,YAGjC,4BAAI,KAAJ,EAAW,YAAX,C,C;MAEiC,YAGlC,4BAAK,KAAL,EAAY,YAAZ,C,C;MAEkC,YAGnC,4BAAM,KAAN,EAAa,YAAb,C,C;MAEmC,YAGpC,4BAAO,KAAP,EAAc,YAAd,C,C;MAEoC,YAGrC,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEqC,YAGtC,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEsC,YAGvC,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAEuC,YAGxC,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAEwC,YAGzC,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAEyC,YAG1C,4BAAa,KAAb,EAAoB,YAApB,C,C;MAE0C,YAG3C,4BAAc,KAAd,EAAqB,YAArB,C,C;MAE2C,YAG5C,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAE4C,YAG7C,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAE6C,YAG9C,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAE8C,YAG/C,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAE+C,YAGhD,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAEgD,YAGjD,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAEiD,YAGlD,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAEkD,YAGnD,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAEmD,YAGpD,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAEoD,YAGrD,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAEqD,YAGtD,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAEsD,YAGvD,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAEuD,YAGxD,4BAAA,KAAA,EAAO,YAAP,C,C;MAEwD,YAGzD,4BAAC,KAAD,EAAQ,YAAR,C,C;MAEyD,YAG1D,4BAAE,KAAF,EAAS,YAAT,C,C;MAE0D,YAG3D,4BAAG,KAAH,EAAU,YAAV,C,C;MAE2D,YAG5D,4BAAI,KAAJ,EAAW,YAAX,C,C;MAE4D,YAG7D,4BAAK,KAAL,EAAY,YAAZ,C,C;MAE6D,YAG9D,4BAAM,KAAN,EAAa,YAAb,C,C;MAE8D,YAG/D,4BAAO,KAAP,EAAc,YAAd,C,C;MAE+D,YAGhE,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEgE,YAGjE,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEiE,YAGlE,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAEkE,YAGnE,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAEmE,YAGpE,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAEoE,YAGrE,4BAAa,KAAb,EAAoB,YAApB,C,C;MAEqE,YAGtE,4BAAc,KAAd,EAAqB,YAArB,C,C;MAEsE,YAGvE,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAEuE,YAGxE,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAEwE,YAGzE,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAEyE,YAG1E,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAE0E,YAG3E,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAE2E,YAG5E,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAE4E,YAG7E,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAE6E,YAG9E,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAE8E,YAG/E,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAE+E,YAGhF,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAEgF,YAGjF,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAEiF,YAGlF,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAEkF,YAGnF,4BAAA,KAAA,EAAO,YAAP,C,C;MAEmF,YAGpF,4BAAC,KAAD,EAAQ,YAAR,C,C;MAEoF,YAGrF,4BAAE,KAAF,EAAS,YAAT,C,C;MAEqF,YAGtF,4BAAG,KAAH,EAAU,YAAV,C,C;MAEsF,YAGvF,4BAAI,KAAJ,EAAW,YAAX,C,C;MAEuF,YAGxF,4BAAK,KAAL,EAAY,YAAZ,C,C;MAEwF,YAGzF,4BAAM,KAAN,EAAa,YAAb,C,C;MAEyF,YAG1F,4BAAO,KAAP,EAAc,YAAd,C,C;MAE0F,YAG3F,4BAAQ,KAAR,EAAe,YAAf,C,C;MAE2F,YAG5F,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAE4F,YAG7F,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAE6F,YAG9F,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAE8F,YAG/F,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAE+F,YAGhG,4BAAa,KAAb,EAAoB,YAApB,C,C;MAEgG,YAGjG,4BAAc,KAAd,EAAqB,YAArB,C,C;MAEiG,YAGlG,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAEkG,YAGnG,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAEmG,YAGpG,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAEoG,YAGrG,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAEqG,YAGtG,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAEsG,YAGvG,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAEuG,YAGxG,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAEwG,YAGzG,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAEyG,YAG1G,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAE0G,YAG3G,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAE2G,YAG5G,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAE4G,YAG7G,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAE6G,YAG9G,4BAAA,KAAA,EAAO,YAAP,C,C;MAE8G,YAG/G,4BAAC,KAAD,EAAQ,YAAR,C,C;MAE+G,YAGhH,4BAAE,KAAF,EAAS,YAAT,C,C;MAEgH,YAGjH,4BAAG,KAAH,EAAU,YAAV,C,C;MAEiH,YAGlH,4BAAI,KAAJ,EAAW,YAAX,C,C;MAEkH,YAGnH,4BAAK,KAAL,EAAY,YAAZ,C,C;MAEmH,YAGpH,4BAAM,KAAN,EAAa,YAAb,C,C;MAEoH,YAGrH,4BAAO,KAAP,EAAc,YAAd,C,C;MAEqH,YAGtH,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEsH,YAGvH,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEuH,YAGxH,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAEwH,YAGzH,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAEyH,YAG1H,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAE0H,YAG3H,4BAAa,KAAb,EAAoB,YAApB,C,C;MAE2H,YAG5H,4BAAc,KAAd,EAAqB,YAArB,C,C;MAE4H,YAG7H,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAE6H,YAG9H,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAE8H,YAG/H,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAE+H,YAGhI,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAEgI,YAGjI,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAEiI,YAGlI,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAEkI,YAGnI,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAEmI,YAGpI,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAEoI,YAGrI,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAEqI,YAGtI,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAEsI,YAGvI,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAEuI,YAGxI,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAEwI,YAGzI,4BAAA,KAAA,EAAO,YAAP,C,C;MAEyI,YAG1I,4BAAC,KAAD,EAAQ,YAAR,C,C;MAE0I,YAG3I,4BAAE,KAAF,EAAS,YAAT,C,C;MAE2I,YAG5I,4BAAG,KAAH,EAAU,YAAV,C,C;MAE4I,YAG7I,4BAAI,KAAJ,EAAW,YAAX,C,C;MAE6I,YAG9I,4BAAK,KAAL,EAAY,YAAZ,C,C;MAE8I,YAG/I,4BAAM,KAAN,EAAa,YAAb,C,C;MAE+I,YAGhJ,4BAAO,KAAP,EAAc,YAAd,C,C;MAEgJ,YAGjJ,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEiJ,YAGlJ,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEkJ,YAGnJ,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAEmJ,YAGpJ,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAEoJ,YAGrJ,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAEqJ,YAGtJ,4BAAa,KAAb,EAAoB,YAApB,C,C;MAEsJ,YAGvJ,4BAAc,KAAd,EAAqB,YAArB,C,C;MAEuJ,YAGxJ,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAEwJ,YAGzJ,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAEyJ,YAG1J,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAE0J,YAG3J,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAE2J,YAG5J,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAE4J,YAG7J,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAE6J,YAG9J,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAE8J,YAG/J,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAE+J,YAGhK,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAEgK,YAGjK,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAEiK,YAGlK,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAEkK,YAGnK,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAEmK,YAGpK,4BAAA,KAAA,EAAO,YAAP,C,C;MAEoK,YAGrK,4BAAC,KAAD,EAAQ,YAAR,C,C;MAEqK,YAGtK,4BAAE,KAAF,EAAS,YAAT,C,C;MAEsK,YAGvK,4BAAG,KAAH,EAAU,YAAV,C,C;MAEuK,YAGxK,4BAAI,KAAJ,EAAW,YAAX,C,C;MAEwK,YAGzK,4BAAK,KAAL,EAAY,YAAZ,C,C;MAEyK,YAG1K,4BAAM,KAAN,EAAa,YAAb,C,C;MAE0K,YAG3K,4BAAO,KAAP,EAAc,YAAd,C,C;MAE2K,YAG5K,4BAAQ,KAAR,EAAe,YAAf,C,C;MAE4K,YAG7K,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAE6K,YAG9K,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAE8K,YAG/K,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAE+K,YAGhL,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAEgL,YAGjL,4BAAa,KAAb,EAAoB,YAApB,C,C;MAEiL,YAGlL,4BAAc,KAAd,EAAqB,YAArB,C,C;MAEkL,YAGnL,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAEmL,YAGpL,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAEoL,YAGrL,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAEqL,YAGtL,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAEsL,YAGvL,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAEuL,YAGxL,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAEwL,YAGzL,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAEyL,YAG1L,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAE0L,YAG3L,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAE2L,YAG5L,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAE4L,YAG7L,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAE6L,YAG9L,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAE8L,YAG/L,4BAAA,KAAA,EAAO,YAAP,C,C;MAE+L,YAGhM,4BAAC,KAAD,EAAQ,YAAR,C,C;MAEgM,YAGjM,4BAAE,KAAF,EAAS,YAAT,C,C;MAEiM,YAGlM,4BAAG,KAAH,EAAU,YAAV,C,C;MAEkM,YAGnM,4BAAI,KAAJ,EAAW,YAAX,C,C;MAEmM,YAGpM,4BAAK,KAAL,EAAY,YAAZ,C,C;MAEoM,YAGrM,4BAAM,KAAN,EAAa,YAAb,C,C;MAEqM,YAGtM,4BAAO,KAAP,EAAc,YAAd,C,C;MAEsM,YAGvM,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEuM,YAGxM,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEwM,YAGzM,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAEyM,YAG1M,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAE0M,YAG3M,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAE2M,YAG5M,4BAAa,KAAb,EAAoB,YAApB,C,C;MAE4M,YAG7M,4BAAc,KAAd,EAAqB,YAArB,C,C;MAE6M,YAG9M,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAE8M,YAG/M,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAE+M,YAGhN,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAEgN,YAGjN,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAEiN,YAGlN,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAEkN,YAGnN,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAEmN,YAGpN,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAEoN,YAGrN,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAEqN,YAGtN,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAEsN,YAGvN,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAEuN,YAGxN,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAEwN,YAGzN,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAEyN,YAG1N,4BAAA,KAAA,EAAO,YAAP,C,C;MAE0N,YAG3N,4BAAC,KAAD,EAAQ,YAAR,C,C;MAE2N,YAG5N,4BAAE,KAAF,EAAS,YAAT,C,C;MAE4N,YAG7N,4BAAG,KAAH,EAAU,YAAV,C,C;MAE6N,YAG9N,4BAAI,KAAJ,EAAW,YAAX,C,C;MAE8N,YAG/N,4BAAK,KAAL,EAAY,YAAZ,C,C;MAE+N,YAGhO,4BAAM,KAAN,EAAa,YAAb,C,C;MAEgO,YAGjO,4BAAO,KAAP,EAAc,YAAd,C,C;MAEiO,YAGlO,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEkO,YAGnO,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEmO,YAGpO,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAEoO,YAGrO,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAEqO,YAGtO,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAEsO,YAGvO,4BAAa,KAAb,EAAoB,YAApB,C,C;MAEuO,YAGxO,4BAAc,KAAd,EAAqB,YAArB,C,C;MAEwO,YAGzO,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAEyO,YAG1O,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAE0O,YAG3O,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAE2O,YAG5O,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAE4O,YAG7O,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAE6O,YAG9O,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAE8O,YAG/O,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAE+O,YAGhP,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAEgP,YAGjP,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAEiP,YAGlP,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAEoP,YAGrP,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAEuP,YAGxP,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAE0P,YAG3P,4BAAA,KAAA,EAAO,YAAP,C,C;MAE6P,YAG9P,4BAAC,KAAD,EAAQ,YAAR,C,C;MAE8P,YAG/P,4BAAE,KAAF,EAAS,YAAT,C,C;MAEgQ,YAGjQ,4BAAG,KAAH,EAAU,YAAV,C,C;MAEmQ,YAGpQ,4BAAI,KAAJ,EAAW,YAAX,C,C;MAEsQ,YAGvQ,4BAAK,KAAL,EAAY,YAAZ,C,C;MAEyQ,YAG1Q,4BAAM,KAAN,EAAa,YAAb,C,C;MAE4Q,YAG7Q,4BAAO,KAAP,EAAc,YAAd,C,C;MAE+Q,YAGhR,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEkR,YAGnR,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEqR,YAGtR,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAEwR,YAGzR,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAE2R,YAG5R,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAE8R,YAG/R,4BAAa,KAAb,EAAoB,YAApB,C,C;MAEiS,YAGlS,4BAAc,KAAd,EAAqB,YAArB,C,C;MAEoS,YAGrS,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAEuS,YAGxS,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAE0S,YAG3S,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAE6S,YAG9S,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAEgT,YAGjT,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAEmT,YAGpT,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAEsT,YAGvT,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAEyT,YAG1T,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAE4T,YAG7T,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAE+T,YAGhU,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAEkU,YAGnU,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAEqU,YAGtU,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAEwU,YAGzU,4BAAA,KAAA,EAAO,YAAP,C,C;MAE2U,YAG5U,4BAAC,KAAD,EAAQ,YAAR,C,C;MAE8U,YAG/U,4BAAE,KAAF,EAAS,YAAT,C,C;MAEiV,YAGlV,4BAAG,KAAH,EAAU,YAAV,C,C;MAEoV,YAGrV,4BAAI,KAAJ,EAAW,YAAX,C,C;MAEuV,YAGxV,4BAAK,KAAL,EAAY,YAAZ,C,C;MAE0V,YAG3V,4BAAM,KAAN,EAAa,YAAb,C,C;MAE6V,YAG9V,4BAAO,KAAP,EAAc,YAAd,C,C;MAEgW,YAGjW,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEmW,YAGpW,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEsW,YAGvW,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAEyW,YAG1W,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAE4W,YAG7W,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAE+W,YAGhX,4BAAa,KAAb,EAAoB,YAApB,C,C;MAEkX,YAGnX,4BAAc,KAAd,EAAqB,YAArB,C,C;MAEqX,YAGtX,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAEwX,YAGzX,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAE2X,YAG5X,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAE8X,YAG/X,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAEiY,YAGlY,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAEoY,YAGrY,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAEuY,YAGxY,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAE0Y,YAG3Y,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAE6Y,YAG9Y,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAEgZ,YAGjZ,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAEmZ,YAGpZ,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAEsZ,YAGvZ,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAEyZ,YAG1Z,4BAAA,KAAA,EAAO,YAAP,C,C;MAE4Z,YAG7Z,4BAAC,KAAD,EAAQ,YAAR,C,C;MAE+Z,YAGha,4BAAE,KAAF,EAAS,YAAT,C,C;MAEkb,YAGnb,4BAAG,KAAH,EAAU,YAAV,C,C;MAEqb,YAGtb,4BAAI,KAAJ,EAAW,YAAX,C,C;MAEwb,YAGzb,4BAAK,KAAL,EAAY,YAAZ,C,C;MAE2b,YAG5b,4BAAM,KAAN,EAAa,YAAb,C,C;MAE8b,YAG/b,4BAAO,KAAP,EAAc,YAAd,C,C;MAEic,YAGlc,4BAAQ,KAAR,EAAe,YAAf,C,C;MAEoc,YAGrc,4BAAS,KAAT,EAAgB,YAAhB,C,C;MAEuc,YAGxc,4BAAU,KAAV,EAAiB,YAAjB,C,C;MAE0c,YAG3c,4BAAW,KAAX,EAAkB,YAAlB,C,C;MAE6c,YAG9c,4BAAY,KAAZ,EAAmB,YAAnB,C,C;MAEgd,YAGjd,4BAAa,KAAb,EAAoB,YAApB,C,C;MAEmd,YAGpd,4BAAc,KAAd,EAAqB,YAArB,C,C;MAEsd,YAGvd,4BAAe,KAAf,EAAsB,YAAtB,C,C;MAEyd,YAG1d,4BAAgB,KAAhB,EAAuB,YAAvB,C,C;MAE4d,YAG7d,4BAAiB,KAAjB,EAAwB,YAAxB,C,C;MAEie,YAGle,4BAAkB,KAAlB,EAAyB,YAAzB,C,C;MAEoe,YAGre,4BAAmB,KAAnB,EAA0B,YAA1B,C,C;MAEue,YAGxe,4BAAoB,KAApB,EAA2B,YAA3B,C,C;MAE0e,YAG3e,4BAAqB,KAArB,EAA4B,YAA5B,C,C;MAE6e,YAG9e,4BAAsB,KAAtB,EAA6B,YAA7B,C,C;MAEgf,YAGjf,4BAAuB,KAAvB,EAA8B,YAA9B,C,C;MAEmf,YAGpf,4BAAwB,KAAxB,EAA+B,YAA/B,C,C;MAEsf,YAGvf,4BAAyB,KAAzB,EAAgC,YAAhC,C,C;MAEyf,YAG1f,4BAA0B,KAA1B,EAAiC,YAAjC,C,C;MAE4f,YAG7f,4BAAA,KAAA,EAAO,YAAP,C,C;MAE+f,YAGhg,4BAAC,KAAD,EAAQ,YAAR,C,C;MAEkgB,YAGngB,4BAAE,KAAF,EAAS,YAAT,C,C;MAEqgB,YAGtgB,4BAAG,KAAH,EAAU,YAAV,C,C;MAEwgB,YAGzgB,4BAAI,KAAJ,EAAW,YAAX,C,C;MAE2gB,YAG5gB,4BAAK,KAAL,EAAY,YAAZ,C,C;MAE8gB,YAG/gB,4BAAM,KAAN,EAAa,YAAb,C,C;MAEihB,YAGlhB,4BAAO,KAAP,EAAc,YAAd,C,C;MAEohB,YAGrhB,4B

,+B;MiBpPA,wB;QAEuD,OASA,eAAM,IAAK,KAAK,KAAW,CjBkP7C,eAAW,oBiB3PiB,KjB2PtB,KAAK,CAAL,UAAN,CiBlP6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MfkPA,aAS6D,0B;MAT7D,+B;MelPA,wB;QAEwD,OAMD,eAAM,IAAK,KAAK,KAAW,CfmP7C,eAAW,oBezPkB,KfyPvB,KAAK,CAAL,YAAN,CenP6C,MAAX,CAAhB,C;O;KARvD,C;4FAGA,yB;MhBkQA,kBAS6D,sB;MAT7D,+B;MgBlQA,wB;QAEsD,OAGC,eAAM,IAAK,KAAK,KAAW,ChBsQ7C,eAAW,oBgBzQgB,KgByQrB,KAAK,CAAL,iBAAN,CgBtQ6C,MAAX,CAAhB,C;O;KALvD,C;4FAGA,yB;MAAA,+B;MAAA,wB;QAEuD,sBAAM,IAAK,KAAK,KAAK,KAAM,KAAX,CAAhB,C;O;KAFvD,C;8FAIA,yB;MjBuOA,WAS6D,wB;MAT7D,+B;MiBvOA,wB;QAEwD,OASA,eAAM,IAAK,KAAK,UAAY,CjBqOpF,C/C,eAAW,oBiBrOiF,KjBqOtF,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0OmB,KjBzOnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBiPmB,KjBhPnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwPmB,KjBvPnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+PmB,KjB9PnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBsQmB,KjBrQnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6QmB,KjB5QnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBoRmB,KjBnRnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB2RmB,KjB1RnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBkSmB,KjBjSnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjByOmB,KjBxOnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBgPmB,KjB/OnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBuPmB,KjBtPnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB8PmB,KjB7PnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBqQmB,KjBpQnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB4QmB,KjB3QnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBmRmB,KjBlRnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0RmB,KjBzRnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBiSmB,KjBhSnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwSmB,KjBvSnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+SmB,KjB9SnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBsTmB,KjBrTnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6TmB,KjB5TnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBoUmB,KjBnUnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB2UmB,KjB1UnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBkVmB,KjBjVnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjByVmB,KjBxVnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBgWmB,KjB/VnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBuWmB,KjBtWnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB8WmB,KjB7WnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBqXmB,KjBpXnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB4XmB,KjB3XnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBmYmB,KjBlYnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0YmB,KjBzYnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBiZmB,KjBhZnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwZmB,KjBvZnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+ZmB,KjB9ZnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBsamB,KjBramB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6amB,KjB5anB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBobmB,KjBnbnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB2bmB,KjB1bnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBkcmB,KjBjcnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBycmB,KjBxcnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBgdmB,KjB/cnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBudmB,KjBtdnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB8dmB,KjB7dnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBqemB,KjBpenB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB4emB,KjB3enB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBmfmB,KjBlfnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0fmB,KjBzfnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBigBmB,KjBhgBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwgBmB,KjBvgBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+gBmB,KjB9gBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBshBmB,KjBrhBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6hBmB,KjB5hBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBoiBmB,KjBniBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB2iBmB,KjB1iBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBkjBmB,KjBjjBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjByjBmB,KjBxjBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBgkBmB,KjB/jBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBukBmB,KjBtkBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB8kBmB,KjB7kBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBqlBmB,KjBplBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB4lBmB,KjB3lBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBmmBmB,KjBlmBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0mBmB,KjBzmBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBinBmB,KjBhnBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwnBmB,KjBvnBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+nBmB,KjB9nBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBspBmB,KjBrpBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6pBmB,KjB5pBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBoqBmB,KjBnqBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB2qBmB,KjB1qBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBkrBmB,KjBjrBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjByrBmB,KjBxrBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBgsBmB,KjB/rBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBusBmB,KjBtsBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB8sBmB,KjB7sBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBqtBmB,KjBptBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB4tBmB,KjB3tBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBmuBmB,KjBluBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0uBmB,KjBzuBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBivBmB,KjBhvBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwvBmB,KjBvvBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+vBmB,KjB9vBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBswBmB,KjBrwBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6wBmB,KjB5wBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBoxBmB,KjBnxBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB2xBmB,KjB1xBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBkyBmB,KjBjyBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjByyBmB,KjBxyBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBgzBmB,KjB/yBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBuzBmB,KjBtzBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB8zBmB,KjB7zBnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBq0BmB,KjBp0BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB40BmB,KjB30BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBm1BmB,KjBl1BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB01BmB,KjBz1BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBi2BmB,KjBh2BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBw2BmB,KjBv2BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+2BmB,KjB92BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBs3BmB,KjBr3BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB63BmB,KjB53BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBo4BmB,KjBn4BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB24BmB,KjB14BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBk5BmB,KjBj5BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBy5BmB,KjBx5BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBg6BmB,KjB/5BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBu6BmB,KjBt6BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB86BmB,KjB76BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBq7BmB,KjBp7BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB47BmB,KjB37BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBm8BmB,KjBl8BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB08BmB,KjBz8BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBi9BmB,KjBh9BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBw9BmB,KjBv9BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+9BmB,KjB99BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBs+BmB,KjBr+BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6+BmB,KjB5+BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBo/BmB,KjBn/BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB2/BmB,KjB1/BnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBkgCmB,KjBjgCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBygCmB,KjBxgCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBghCmB,KjB/gCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBuhCmB,KjBthCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB8hCmB,KjB7hCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBqiCmB,KjBpiCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB4iCmB,KjB3iCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBmjCmB,KjBljCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0jCmB,KjBzjCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBikCmB,KjBhkCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwkCmB,KjBvkCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+kCmB,KjB9kCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBslCmB,KjBrlCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6lCmB,KjB5lCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBomCmB,KjBnmCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB2mCmB,KjB1mCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBknCmB,KjBjnCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBynCmB,KjBxnCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBgoCmB,KjB/nCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBuoCmB,KjBtoCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB8oCmB,KjB7oCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBqpCmB,KjBppCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB4pCmB,KjB3pCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBmqCmB,KjBlqCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0qCmB,KjBzqCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBirCmB,KjBhrCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwrCmB,KjBvrCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+rCmB,KjB9rCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBssCmB,KjBrsCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6sCmB,KjB5sCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBotCmB,KjBntCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB2tCmB,KjB1tCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBkuCmB,KjBjuCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjByuCmB,KjBxuCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBgvCmB,KjB/uCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBuvCmB,KjBtvCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB8vCmB,KjB7vCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBqwCmB,KjBpwCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB4wCmB,KjB3wCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBmxCmB,KjBlxCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0xCmB,KjBzxCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBiyCmB,KjBhyCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwyCmB,KjBvyCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+yCmB,KjB9yCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBszCmB,KjBrzCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6zCmB,KjB5zCnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBo0CmB,KjBn0CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB20CmB,KjB10CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBk1CmB,KjBj1CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBy1CmB,KjBx1CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBg2CmB,KjB/1CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBu2CmB,KjBt2CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB82CmB,KjB72CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBq3CmB,KjBp3CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB43CmB,KjB33CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBm4CmB,KjBl4CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB04CmB,KjBz4CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBi5CmB,KjBh5CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBw5CmB,KjBv5CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+5CmB,KjB95CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBs6CmB,KjBr6CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB66CmB,KjB56CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBo7CmB,KjBn7CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB27CmB,KjB17CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBk8CmB,KjBj8CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBy8CmB,KjBx8CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBg9CmB,KjB/8CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBu9CmB,KjBt9CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB89CmB,KjB79CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBq+CmB,KjBp+CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB4+CmB,KjB3+CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBm/CmB,KjBl/CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0/CmB,KjBz/CnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBigDmB,KjBhgDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwgDmB,KjBvgDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+gDmB,KjB9gDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBshDmB,KjBrhDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB6hDmB,KjB5hDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBoiDmB,KjBniDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB2iDmB,KjB1iDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBkjDmB,KjBjjDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjByjDmB,KjBxjDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBgkDmB,KjB/jDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBukDmB,KjBtkDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB8kDmB,KjB7kDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBqlDmB,KjBplDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB4lDmB,KjB3lDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBmmDmB,KjBlmDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB0mDmB,KjBzmDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBinDmB,KjBhnDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBwnDmB,KjBvnDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjB+nDmB,KjB9nDnB,KAAK,CAAL,UAAN,CiBrO6C,MAAX,CAAhB,C;O;KAXvD,C;4FAGA,yB;MjBspDmB,KjBrpDnB,KAAK,CAAL,UAAN

ASA,yB;MAAA,+B;MAAA,wB;QAEmD,sBAAM,IAAK,KAAL,KAAc,KAAM,KAApB,CAAN,C;O;KAFnD,C;wFAGA,yB;MAAA,+B;MAAA,wB;QAEkD,sBAAM,IAAK,KAAL,IAAa,KAAM,KAAnB,CAAN,C;O;KAFlD,C;0FAGA,yB;MAAA,+B;MAAA,wB;QAEmD,sBAAM,IAAK,KAAL,KAAc,KAAM,KAApB,CAAN,C;O;KAFnD,C;0EAGA,yB;MAAA,+B;MAAA,mB;QAEiC,sBAAM,SAAK,MAAX,C;O;KAFjC,C;gFAIA,yB;MAAA,0B;MAAA,mB;QAUmC,OAAK,OAAL,SAAK,S;O;KAVxC,C;kFAWA,yB;MAAA,4B;MAAA,mB;QAUqC,OAAK,QAAL,SAAK,S;O;KAV1C,C;8EAWA,Y;MAUiC,OAAA,SAAK,Q;K;gFAGtC,C,Y;MASmC,gB;K;kFAEnC,yB;MjBmEJ,0B;MAAA,+B;MiBnEI,mB;QASqC,OjBqEC,eAAW,OiBrEZ,SjBqEY,SAAX,C;O;KiB9EtC,C;oFAUA,yB;Mf0DJ,4B;MAAA,iC;Me1DI,mB;QASuC,Of4DC,gBAAY,Qe5Db,Sf4Da,SAAZ,C;O;KerExC,C;gFAUA,yB;MhBqEJ,6B;MgBrEI,mB;QASmC,OhBuEC,cgBvED,ShBuEW,QAAV,C;O;KgBhFpC,C;kFAUA,Y;MAEqC,W;K;kFAErC,yB;MASA,kD;MATA,mB;QAQqC,OASE,cAAc,SAAd,C;O;KAjBvC,C;oFAUA,yB;MAAA,kD;MAAA,mB;QAQuC,OASE,cAAc,SAAd,C;O;KARvC,C;+BAUA,Y;MAAyC,qBAAc,SAAd,C;K;;;;;+BAWzC,Y;MAAc,c;MAG4D,qD;MAH5D,a;K;6BAAA,iB;MAAA,2IAG4D,oCAH5D,G;K;wEAAuW,yB;MAAA,+B;MAAA,4B;QAAW0C,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2C,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4C,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6C,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8C,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+C,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgD,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiD,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkD,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmD,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoD,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqD,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsD,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuD,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwD,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyD,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0D,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2D,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4D,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6D,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8D,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+D,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgE,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiE,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkE,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmE,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoE,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqE,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsE,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuE,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwE,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyE,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0E,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2E,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4E,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6E,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8E,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+E,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgF,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiF,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkF,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmF,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoF,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqF,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsF,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuF,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwF,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyF,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0F,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2F,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4F,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6F,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8F,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+F,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgG,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiG,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkG,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmG,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoG,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqG,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsG,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuG,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwG,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyG,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0G,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2G,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4G,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6G,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8G,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+G,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgH,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiH,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkH,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmH,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoH,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqH,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsH,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuH,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwH,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyH,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0H,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2H,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4H,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6H,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8H,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+H,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgI,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiI,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkI,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmI,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoI,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqI,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsI,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuI,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwI,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyI,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0I,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2I,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4I,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6I,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8I,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+I,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgJ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiJ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkJ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmJ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoJ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqJ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsJ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuJ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwJ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyJ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0J,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2J,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4J,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6J,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8J,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+J,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgK,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiK,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkK,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmK,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoK,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqK,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsK,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuK,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwK,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyK,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0K,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2K,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4K,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6K,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8K,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+K,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgL,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiL,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkL,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmL,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoL,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqL,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsL,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuL,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwL,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyL,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0L,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2L,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4L,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6L,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8L,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+L,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgM,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiM,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkM,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmM,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoM,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqM,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsM,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuM,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwM,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyM,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0M,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2M,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4M,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6M,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8M,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+M,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgN,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiN,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkN,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmN,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoN,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqN,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsN,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuN,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwN,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyN,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0N,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2N,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4N,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6N,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8N,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+N,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgO,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiO,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkO,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmO,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoO,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqO,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsO,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuO,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwO,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyO,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0O,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2O,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4O,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6O,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8O,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+O,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgP,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiP,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkP,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmP,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoP,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqP,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsP,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuP,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwP,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyP,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0P,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2P,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4P,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6P,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW8P,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW+P,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWgQ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWiQ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWkQ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWmQ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWoQ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWqQ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWsQ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWuQ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWwQ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAWyQ,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW0Q,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW2Q,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW4Q,sBAAW,oBAAX,SAAW,CAAX,C;O;KAAlX,C;0FAAA,yB;MAAA,+B;MAAA,4B;QAAW6Q,sBAA

B+MnKG,E/MmKH,CAAN,CAUyB,MAApB,CAAN,CAqClB,MAAK,Q+MlNJ,Q/MkND,CArCkB,U+M7K0B,S/M6Kf,KAAL,KAAoB,CAVzB,U+MnKmC,S/MmK7B,yB+MnKsC,E/MmKtC,CAAN,CAUyB,MAApB,CAAN,CAqClB,MAAK,Q+MlNJ,IAAN,SAAqF,cAAU,6BAAU,EAAV,CAAV,CAAyB,QAA9G,I;K;0CAE5B,Y;MAAkC,OAAI,uBAAO,CAAX,CAAa,GAAgB,UAAF,qBAAU,SAAV,CAAoB,cAAqB,SAAX,rB,WAAW,GAAgD,UAAF,2BAAgB,SAAhB,CAA0B,cAA6B,SAAnB,hB,W;K;IAAhF,sC;MAAA,0C;K;mEAAI,sC;MAAQ,F,4BAAiB,UAAjB,qB,EAAiC,QAAjC,6B,EAAuC,UAAvC,uC,IAAuC,IAAvC,gB,EAAuC,QAAvC,6B,EAAiD,UAAjD,uC,IAAiD,IAAjD,gB,EAAiD,QAAjD,6B,EAA2D,UAA3D,uC,IAA2D,IAA3D,gB,EAA2D,QAA3D,6B,EAAqE,UAArE,uC,IAAqE,IAArE,gB,EAAqE,QAArE,6B,EAA+E,UAA/E,uC,IAA+E,IAA/E,gB,EAA+E,QAA/E,6B,EAAyF,UAAzF,uC,IAAyF,IAAzF,gB,EAAyF,QAAzF,6B,EAAmG,UAAnG,uC,IAAmG,IAAnG,gB,EAAmG,QAAnG,6B,EAA6G,UAA7G,uC,IAA6G,IAA7G,gB,EAA6G,QAA7G,6B,EAAuH,UAAvH,uC,IAAuH,IAAvH,gB,EAAuH,QAAvH,6B,EAAiI,UAAjI,uC,IAAiI,IAAjI,gB,EAAiI,QAAjI,6B,EAA2I,UAA3I,uC,IAA2I,IAA3I,gB,EAA2I,QAA3I,6B,EAAqJ,UAArJ,uC,IAAqJ,IAArJ,gB,EAAqJ,QAArJ,6B,EAA+J,UAA/J,uC,IAA+J,IAA/J,gB,EAA+J,QAA/J,6B,EAAyK,UAAzK,uC,IAAyK,IAAzK,gB,EAAyK,QAAzK,6B,EAAmL,UAAnL,uC,IAAmL,IAAnL,gB,EAAmL,QAAnL,6B,EAA6L,UAA7L,uC,IAA6L,IAA7L,gB,EAA6L,QAA7L,6B,EAAuM,UAAvM,uC,IAAuM,IAAvM,gB,EAAuM,QAAvM,6B,EAAiN,UAAjN,uC,IAAiN,IAAjN,gB,EAAiN,QAAjN,6B,EAA2N,UAA3N,uC,IAA2N,IAA3N,gB,EAA2N,QAA3N,6B,EAAqO,UAArO,uC,IAAqO,IAArO,gB,EAAqO,QAArO,6B,EAA+O,UAA/O,uC,IAA+O,IAA/O,gB,EAA+O,QAA/O,6B,EAAyP,UAAzP,uC,IAAyP,IAAzP,gB,EAAyP,QAAzP,6B,EAAmQ,UAAnQ,uC,IAAmQ,IAAnQ,gB,EAAmQ,QAAnQ,6B,EAA6Q,UAA7Q,uC,IAA6Q,IAA7Q,gB,EAA6Q,QAA7Q,6B,EAAuR,UAAvR,uC,IAAuR,IAAvR,gB,EAAuR,QAAvR,6B,EAAiS,UAAjS,uC,IAAiS,IAAjS,gB,EAAiS,QAAjS,6B,EAA2S,UAA3S,uC,IAA2S,IAA3S,gB,EAA2S,QAA3S,6B,EAAqT,UAArT,uC,IAAqT,IAArT,gB,EAAqT,QAArT,6B,EAA+T,UAA/T,uC,IAA+T,IAA/T,gB,EAA+T,QAA/T,6B,EAAyU,UAAzU,uC,IAAyU,IAAzU,gB,EAAyU,QAAzU,6B,EAAmV,UAAnV,uC,IAAmV,IAAnV,gB,EAAmV,QAAnV,6B,EAA6V,UAA7V,uC,IAA6V,IAA7V,gB,EAA6V,QAA7V,6B,EAAuW,UAAvW,uC,IAAuW,IAAvW,gB,EAAuW,QAAvW,6B,EAAiX,UAAjX,uC,IAAiX,IAAjX,gB,EAAiX,QAAjX,6B,EAA2X,UAA3X,uC,IAA2X,IAA3X,gB,EAA2X,QAA3X,6B,EAAqY,UAArY,uC,IAAqY,IAArY,gB,EAAqY,QAArY,6B,EAA+Y,UAA/Y,uC,IAA+Y,IAA/Y,gB,EAA+Y,QAA/Y,6B,EAAyZ,UAAzZ,uC,IAAyZ,IAAzZ,gB,EAAyZ,QAAzZ,6B,EAAma,UAAna,uC,IAAma,IAAna,gB,EAAma,QAAna,6B,EAA6a,UAA7a,uC,IAA6a,IAA7a,gB,EAA6a,QAA7a,6B,EAAub,UAAvb,uC,IAAub,IAAvb,gB,EAAub,QAAvb,6B,EAAic,UAAjc,uC,IAAic,IAAjc,gB,EAAic,QAAjc,6B,EAA2c,UAA3c,uC,IAA2c,IAA3c,gB,EAA2c,QAA3c,6B,EAAqd,UAArd,uC,IAAqd,IAArd,gB,EAAqd,QAArd,6B,EAA+d,UAA/d,uC,IAA+d,IAA/d,gB,EAA+d,QAA/d,6B,EAAye,UAAze,uC,IAAye,IAAze,gB,EAAye,QAAze,6B,EAAmf,UAAnf,uC,IAAmf,IAAnf,gB,EAAmf,QAAnf,6B,EAA6f,UAA7f,uC,IAA6f,IAA7f,gB,EAA6f,QAA7f,6B,EAAugB,UAAvgB,uC,IAAugB,IAAvgB,gB,EAAugB,QAAvgB,6B,EAAihB,UAAjhB,uC,IAAihB,IAAjhB,gB,EAAihB,QAAjhB,6B,EAA2hB,UAA3hB,uC,IAA2hB,IAA3hB,gB,EAA2hB,QAA3hB,6B,EAAqiB,UAAriB,uC,IAAqiB,IAAriB,gB,EAAqiB,QAAriB,6B,EAA+iB,UAA/iB,uC,IAA+iB,IAA/iB,gB,EAA+iB,QAA/iB,6B,EAAyjB,UAAzjB,uC,IAAyjB,IAAzjB,gB,EAAyjB,QAAzjB,6B,EAAmkB,UAAnkB,uC,IAAmkB,IAAnkB,gB,EAAmkB,QAAnkB,6B,EAA6kB,UAA7kB,uC,IAA6kB,IAA7kB,gB,EAA6kB,QAA7kB,6B,EAAumB,UAAvmB,uC,IAAumB,IAAvmB,gB,EAAumB,QAAvmB,6B,EAAinB,UAAjnB,uC,IAAinB,IAAjnB,gB,EAAinB,QAAjnB,6B,EAA2nB,UAA3nB,uC,IAA2nB,IAA3nB,gB,EAA2nB,QAA3nB,6B,EAAqoB,UAAroB,uC,IAAqoB,IAAroB,gB,EAAqoB,QAAroB,6B,EAA+oB,UAA/oB,uC,IAA+oB,IAA/oB,gB,EAA+oB,QAA/oB,6B,EAAyqB,UAAzqB,uC,IAAyqB,IAAzqB,gB,EAAyqB,QAAzqB,6B,EAAmrB,UAAnrB,uC,IAAmrB,IAAnrB,gB,EAAmrB,QAAnrB,6B,EAA6rB,UAA7rB,uC,IAA6rB,IAA7rB,gB,EAA6rB,QAA7rB,6B,EAAutB,UAAvtB,uC,IAAutB,IAAvtB,gB,EAAutB,QAAvtB,6B,EAAiuB,UAAjuB,uC,IAAiuB,IAAjuB,gB,EAAiuB,QAAjuB,6B,EAA2uB,UAA3uB,uC,IAA2uB,IAA3uB,gB,EAA2uB,QAA3uB,6B,EAAqvB,UAArvB,uC,IAAqvB,IAArvB,gB,EAAqvB,QAArvB,6B,EAA+vB,UAA/vB,uC,IAA+vB,IAA/vB,gB,EAA+vB,QAA/vB,6B,EAAyxB,UAAzxB,uC,IAAyxB,IAAzxB,gB,EAAyxB,QAAzxB,6B,EAAmyB,UAAnyB,uC,IAAmyB,IAAnyB,gB,EAAmyB,QAAnyB,6B,EAA6yB,UAA7yB,uC,IAA6yB,IAA7yB,gB,EAA6yB,QAA7yB,6B,EAAuzB,UAAvzB,uC,IAAuzB,IAAvzB,gB,EAAuzB,QAAvzB,6B,EAAi0B,UAAj0B,uC,IAAi0B,IAAj0B,gB,EAAi0B,QAAj0B,6B,EAA20B,UAA30B,uC,IAA20B,IAA30B,gB,EAA20B,QAA30B,6B,EAAq1B,UAAr1B,uC,IAAq1B,IAAr1B,gB,EAAq1B,QAAr1B,6B,EAA+1B,UAA/1B,uC,IAA+1B,IAA/1B,gB,EAA+1B,QAA/1B,6B,EAAy2B,UAAz2B,uC,IAAy2B,IAAz2B,gB,EAAy2B,QAAz2B,6B,EAAm3B,UAAn3B,uC,IAAm3B,IAAn3B,gB,EAAm3B,QAAn3B,6B,EAA63B,UAA73B,uC,IAA63B,IAA73B,gB,EAA63B,QAA73B,6B,EAAu4B,UAAv4B,uC,IAAu4B,IAAv4B,gB,EAAu4B,QAAv4B,6B,EAAi5B,UAAj5B,uC,IAAi5B,IAAj5B,gB,EAAi5B,QAAj5B,6B,EAA25B,UAA35B,uC,IAA25B,IAA35B,gB,EAA25B,QAA35B,6B,EAAq6B,UAAr6B,uC,IAAq6B,IAAr6B,gB,EAAq6B,QAAr6B,6B,EAA+6B,UAA/6B,uC,IAA+6B,IAA/6B,gB,EAA+6B,QAA/6B,6B,EAAy7B,UAAz7B,uC,IAAy7B,IAAz7B,gB,EAAy7B,QAAz7B,6B,EAAm8B,UAAn8B,uC,IAAm8B,IAAn8B,gB,EAAm8B,QAAn8B,6B,EAA68B,UAA78B,uC,IAA68B,IAA78B,gB,EAA68B,QAA78B,6B,EAAu9B,UAAv9B,uC,IAAu9B,IAAv9B,gB,EAAu9B,QAAv9B,6B,EAAi+B,UAAj+B,uC,IAAi+B,IAAj+B,gB,EAAi+B,QAAj+B,6B,EAA2+B,UAA3+B,uC,IAA2+B,IAA3+B,gB,EAA2+B,QAA3+B,6B,EAAqgC,UAArgC,uC,IAAqgC,IAArgC,gB,EAAqgC,QAArgC,6B,EAA+gC,UAA/gC,uC,IAA+gC,IAA/gC,gB,EAA+gC,QAA/gC,6B,EAAyhC,UAAzhC,uC,IAAyhC,IAAzhC,gB,EAAyhC,QAAzhC,6B,EAAmiC,UAAniC,uC,IAAmiC,IAAniC,gB,EAAmiC,QAAniC,6B,EAA6iC,UAA7iC,uC,IAA6iC,IAA7iC,gB,EAA6iC,QAA7iC,6B,EAAukC,UAAvkC,uC,IAAukC,IAAvkC,gB,EAAukC,QAAvkC,6B,EAAilC,UAAjlC,uC,IAAilC,IAAjlC,gB,EAAilC,QAAjlC,6B,EAA2lC,UAA3lC,uC,IAA2lC,IAA3lC,gB,EAA2lC,QAA3lC,6B,EAAqmC,UAArmC,uC,IAAqmC,IAArmC,gB,EAAqmC,QAArmC,6B,EAA+mC,UAA/mC,uC,IAA+mC,IAA/mC,gB,EAA+mC,QAA/mC,6B,EAAyoC,UAAzoC,uC,IAAyoC,IAAzoC,gB,EAAyoC,QAAzoC,6B,EAAmpC,UAAnpC,uC,IAAmpC,IAAnpC,gB,EAAmpC,QAAnpC,6B,EAA6pC,UAA7pC,uC,IAA6pC,IAA7pC,gB,EAA6pC,QAA7pC,6B,EAAuqC,UAAvqC,uC,IAAuqC,IAAvqC,gB,EAAuqC,QAAvqC,6B,EAAirC,UAAjrC,uC,IAAirC,IAAjrC,gB,EAAirC,QAAjrC,6B,EAA2rC,UAA3rC,uC,IAA2rC,IAA3rC,gB,EAA2rC,QAA3rC,6B,EAAqsC,UAArsC,uC,IAAqsC,IAArsC,gB,EAAqsC,QAArsC,6B,EAA+sC,UAA/sC,uC,IAA+sC,IAA/sC,gB,EAA+sC,QAA/sC,6B,EAAyuC,UAAzuC,uC,IAAyuC,IAAzuC,gB,EAAyuC,QAAzuC,6B,EAAmvC,UAAnvC,uC,IAAmvC,IAAnvC,gB,EAAmvC,QAAnvC,6B,EAA6vC,UAA7vC,uC,IAA6vC,IAA7vC,gB,EAA6vC,QAA7vC,6B,EAAuwC,UAAvwC,uC,IAAuwC,IAAvwC,gB,EAAuwC,QAAvwC,6B,EAAixC,UAAjxC,uC,IAAixC,IAAjxC,gB,EAAixC,QAAjxC,6B,EAA2xC,UAA3xC,uC,IAA2xC,IAA3xC,gB,EAA2xC,QAA3xC,6B,EAAqyC,UAAryC,uC,IAAqyC,IAAryC,gB,EAAqyC,QAAryC,6B,EAA+yC,UAA/yC,uC,IAA+yC,IAA/yC,gB,EAA+yC,QAA/yC,6B,EAAy0C,UAAz0C,uC,IAAy0C,IAAz0C,gB,EAAy0C,QAAz0C,6B,EAAm1C,UAAn1C,uC,IAAm1C,IAAn1C,gB,EAAm1C,QAAn1C,6B,EAA61C,UAA71C,uC,IAA61C,IAA71C,gB,EAA61C,QAA71C,6B,EAAu2C,UAAv2C,uC,IAAu2C,IAAv2C,gB,EAAu2C,QAAv2C,6B,EAAi3C,UAAj3C,uC,IAAi3C,IAAj3C,gB,EAAi3C,QAAj3C,6B,EAA23C,UAA33C,uC,IAA23C,IAA33C,gB,EAA23C,QAA33C,6B,EAAq4C,UAAr4C,uC,IAAq4C,IAAr4C,gB,EAAq4C,QAAr4C,6B,EAA+4C,UAA/4C,uC,IAA+4C,IAA/4C,gB,EAA+4C,QAA/4C,6B,EAAy5C,UAAz5C,uC,IAAy5C,IAAz5C,gB,EAAy5C,QAAz5C,6B,EAAm6C,UAAn6C,uC,IAAm6C,IAAn6C,gB,EAAm6C,QAAn6C,6B,EAA66C,UAA76C,uC,IAA66C,IAA76C,gB,EAA66C,QAA76C,6B,EAAu7C,UAAv7C,uC,IAAu7C,IAAv7C,gB,EAAu7C,QAAv7C,6B,EAAi8C,UAAj8C,uC,IAAi8C,IAAj8C,gB,EAAi8C,QAAj8C,6B,EAA28C,UAA38C,uC,IAA28C,IAA38C,gB,EAA28C,QAA38C,6B,EAAq9C,UAAr9C,uC,IAAq9C,IAAr9C,gB,EAAq9C,QAAr9C,6B,EAA+9C,UAA/9C,uC,IAA+9C,IAA/9C,gB,EAA+9C,QAA/9C,6B,EAAy+C,UAAz+C,uC,IAAy+C,IAAz+C,gB,EAAy+C,QAAz+C,6B,EAAm/C,UAAn/C,uC,IAAm/C,IAAn/C,gB,EAAm/C,QAAn/C,6B,EAA6/C,UAA7/C,uC,IAA6/C,IAA7/C,gB,EAA6/C,QAA7/C,6B,EAAuhD,UAAvhD,uC,IAAuhD,IAAvhD,gB,EAAuhD,QAAvhD,6B,EAAiiD,UAAjiD,uC,IAAiiD,IAAjiD,gB,EAAiiD,QAAjiD,6B,EAA2iD,UAA3iD,uC,IAA2iD,IAA3iD,gB,EAA2iD,QAA3iD,6B,EAAqjD,UAArjD,uC,IAAqjD,IAArjD,gB,EAAqjD,QAArjD,6B,EAA+jD,UAA/jD,uC,IAA+jD,IAA/jD,gB,EAA+jD,QAA/jD,6B,EAAylD,UAAzlD,uC,IAAylD,IAAzlD,gB,EAAylD,QAAzlD,6B,EAAmmD,UAAnmD,uC,IAAmmD,IAAnmD,gB,EAAmmD,QAAnmD,6B,EAA6mD,UAA7mD,uC,IAA6mD,IAA7mD,gB,EAA6mD,QAA7mD,6B,EAAuoD,UAAvoD,uC,IAAuoD,IAAvoD,gB,EAAuoD,QAAvoD,6B,EAAipD,UAAjpD,uC,IAAipD,IAAjpD,gB,EAAipD,QAAjpD,6B,EAA2pD,UAA3pD,uC,IAA2pD,IAA3pD,gB,EAA2pD,QAA3pD,6B,EAAqqD,UAArqD,uC,IAAqqD,IAArqD,gB,EAAqqD,QAArqD,6B,EAA+qD,UAA/qD,uC,IAA+qD,IAA/qD,gB,EAA+qD,QAA/qD,6B,EAAysD,UAAzsD,uC,IAAysD,IAAzsD,gB,EAAysD,QAAzsD,6B,EAAmtD,UAAntD,uC,IAAmtD,IAAntD,gB,EAAmtD,QAAntD,6B,EAA6tD,UAA7tD,uC,IAA6tD,IAA7tD,gB,EAA6tD,QAA7tD,6B,EAAuvD,UAAvvD,uC,IAAuvD,IAAvvD,gB,EAAuvD,QAAvvD,6B,EAAiwD,UAAjwD,uC,IAAiwD,IAAjwD,gB,EAAiwD,QAAjwD,6B,EAA2wD,UAA3wD,uC,IAA2wD,IAA3wD,gB,EAA2wD,QAA3wD,6B,EAAqxD,UAArxD,uC,IAAqxD,IAArxD,gB,EAAqxD,QAArxD,6B,EAA+xD,UAA/xD,uC,IAA+xD,IAA/xD,gB,EAA+xD,QAA/xD,6B,EAAyzD,UAAzzD,uC,IAAyzD,IAAzzD,gB,EAAyzD,QAAzzD,6B,EAAm0D,UAAn0D,uC,IAAm0D,IAAn0D,gB,EAAm0D,QAAn0D,6B,EAA60D,UAA70D,uC,IAA60D,IAA70D,gB,EAA60D,QAA70D,6B,EAAu1D,UAAv1D,uC,IAAu1D,IAAv1D,gB,EAAu1D,QAAv1D,6B,EAAi2D,UAAj2D,uC,IAAi2D,IAAj2D,gB,EAAi2D,QAAj2D,6B,EAA22D,UAA32D,uC,IAA22D,IAA32D,gB,EAA22D,QAA32D,6B,EAAq3D,UAAr3D,uC,IAAq3D,IAAr3D,gB,EAAq3D,QAAr3D,6B,EAA+3D,UAA/3D,uC,IAA+3D,IAA/3D,gB,EAA+3D,QAA/3D,6B,EAAy4D,UAAz4D,uC,IAAy4D,IAAz4D,gB,EAAy4D,QAAz4D,6B,EAAm5D,UAAn5D,uC,IAAm5D,IAAn5D,gB,EAAm5D,QAAn5D,6B,EAA65D,UAA75D,uC,IAA65D,IAA75D,gB,EAA65D,QAA75D,6B,EAAu6D,UAAv6D,uC,IAAu6D,IAAv6D,gB,EAAu6D,QAAv6D,6B,EAAi7D,UAAj7D,uC,IAAi7D,IAAj7D,gB,EAAi7D,QAAj7D,6B,EAA27D,UAA37D,uC,IAA27D,IAA37D,gB,EAA27D,QAA37D,6B,EAAq8D,UAAr8D,uC,IAAq8D,IAAr8D,gB,EAAq8D,QAAr8D,6B,EAA+8D,UAA/8D,uC,IAA+8D,IAA/8D,gB,EAA+8D,QAA/8D,6B,EAAy9D,UAAz9D,uC,IAAy9D,IAAz9D,gB,EAAy9D,QAAz9D,6B,EAAm+D,UAAn+D,uC,IAAm+D,IAAn+D,gB,EAAm+D,QAAn+D,6B,EAA6+D,UAA7+D,uC,IAA6+D,IAA7+D,gB,EAA6+D,QAA7+D,6B,EAAuhE,UAAvhE,uC,IAAuhE,IAAvhE,gB,EAAuhE,QAAvhE,6B,EAAiiE,UAAjiE,uC,IAAiiE,IAAjiE,gB,EAAiiE,QAAjiE,6B,EAA2iE,UAA3iE,uC,IAA2iE,IAA3iE,gB,EAA2iE,QAA3iE,6B,EAAqjE,UAArjE,uC,IAAqjE,IAArjE,gB,EAAqjE,QAArjE,6B,EAA+jE,UAA/jE,uC,IAA+jE,IAA/jE,gB,EAA+jE,QAA/jE,6B,EAAylE,UAAzlE,uC,IAAylE,IAAzlE,gB,EAAylE,QAAzlE,6B,EAAmmE,UAAnmE,uC,IAAmmE,IAAnmE,gB,EAAmmE,QAAnmE,6B,EAA6mE,UAA7mE,uC,IAA6mE,IAA7mE,gB,EAA6mE,QAA7mE,6B,EAAuoE,UAAvoE,uC,IAAuoE,IAAvoE,gB,EAAuoE,QAAvoE,6B,EAAipE,UAAjpE,uC,IAAipE,IAAjpE,gB,EAAipE,QAAjpE,6B,EAA2pE,UAA3pE,uC,IAA2pE,IAA3pE,gB,EAA2pE,QAA3pE,6B,EAAqqE,UAArqE,uC,IAAqqE,IAArqE,gB,EAAqqE,QAArqE,6B,EAA+qE,UAA/qE,uC,IAA+qE,IAA/qE,gB,EAA+qE,QAA/qE,6B,EAAysE,UAAzsE,uC,IAAysE,IAAzsE,gB,EAAysE,QAAzsE,6B,EAAmtE,UAAntE,uC,IAAmtE,IAAntE,gB,EAAmtE,QAAntE,6B,EAA6tE,UAA7tE,uC,IAA6tE,IAA7tE,gB,EAA6tE,QAA7tE,6B,EAAuvE,UAAvvE,uC,IAAuvE,IAAvvE,gB,EAAuvE,QAAvvE,6B,EAAiwE,UAAjwE,uC,IAAiwE,IAAjwE,gB,EAAiwE,QAAjwE,6B,EAA2wE,UAA3wE,uC,IAA2wE,IAA3wE,gB,EAA2wE,QAA3wE,6B,EAAqxE,UAArxE,uC,IAAqxE,IAArxE,gB,EAAqxE,QAArxE,6B,EAA+xE,UAA/xE,uC,IAA+xE,IAA/xE,gB,EAA+xE,QAA/xE,6B,EAAyzE,UAAzzE,uC,IAAyzE,IAAzzE,gB,EAAyzE,QAAzzE,6B,EAAm0E,UAAn0E,uC,IAAm0E,IAAn0E,gB,EAAm0E,QAAn0E,6B,EAA60E,UAA70E,uC,IAA60E,IAA70E,gB,EAA60E,QAA70E,6B,EAAu1E,UAAv1E,uC,IAAu1E,IAAv1E,gB,EAAu1E,QAAv1E,6B,EAAi2E,UAAj2E,uC,IAAi2E,IAAj2E,gB,EAAi2E,QAAj2E,6B,EAA22E,UAA32E,uC,IAA22E,IAA32E,gB,EAA22E,QAA32E,6B,EAAq3E,UAAr3E,uC,IAAq3E,IAAr3E,gB,EAAq3E,QAAr3E,6B,EAA+3E,UAA/3E,uC,IAA+3E,IAA/3E,gB,EAA+3E,QAA/3E,6B,EAAy4E,UAAz4E,uC,IAAy4E,IAAz4E,gB,EAAy4E,QAAz4E,6B,EAAm5E,UAAn5E,uC,IAAm5E,IAAn5E,gB,EAAm5E,QAAn5E,6B,EAA65E,UAA75E,uC,IAA65E,IAA75E,gB,EAA65E,QAA75E,6B,EAAu6E,UAAv6E,uC,IAAu6E,IAAv6E,gB,EAAu6E,QAAv6E,6B,EAAi7E,UAAj7E,uC,IAAi7E,IAAj7E,gB,EAAi7E,QAAj7E,6B,EAA27E,UAA37E,uC,IAA27E,IAA37E,gB,EAA27E,QAA37E,6B,EAAq8E,UAAr8E,uC,IAAq8E,IAAr8E,gB,EAAq8E,QAAr8E,6B,EAA+8E,UAA/8E,uC,IAA+8E,IAA/8E,gB,EAA+8E,QAA/8E,6B,EAAy9E,UAAz9E,uC,IAAy9E,IAAz9E,gB,EAAy9E,QAAz9E,6B,EAAm+E,UAAn+E,uC,IAAm+E,IAAn+E,gB,EAAm+E,QAAn+E,6B,EAA6+E,UAA7+E,uC,IAA6+E,IAA7+E,gB,EAA6+E,QAA7+E,6B,EAAuhF,UAAvhF,uC,IAAuhF,IAAvhF,gB,EAAuhF,QAAvhF,6B,EAAiiF,UAAjiF,uC,IAAiiF,IAAjiF,gB,EAAiiF,QAAjiF,6B,EAA2iF,UAA3iF,uC,IAA2iF,IAA3iF,gB,EAA2iF,QAA3iF,6B,EAAqjF,UAArjF,uC,IAAqjF,IAArjF,gB,EAAqjF,QAArjF,6B,EAA+jF,UAA/jF,uC,IAA+jF,IAA/jF,gB,EAA+jF,QAA/jF,6B,EAAylF,UAAzlF,uC,IAAylF,IAAzlF,gB,EAAylF,QAAzlF,6B,EAAmmF,UAAnmF,uC,IAAmmF,IAAnmF,gB,EAAmmF,QAAnmF,6B,EAA6mF,UAA7mF,uC,IAA6mF,IAA7mF,gB,EAA6mF,QAA7mF,6B,EAAuoF,UAAvoF,uC,IAAuoF,IAAvoF,gB,EAAuoF,QAAvoF,6B,EAAipF,UAAjpF,uC,IAAipF,IAAjpF,gB,EAAipF,QAAjpF,6B,EAA2pF,UAA3pF,uC,IAA2pF,IAA3pF,gB,EAA2pF,QAA3pF,6B,EAAqqF,UAArqF,uC,IAAqqF,IAArqF,gB,EAAqqF,QAArqF,6B,EAA+qF,UAA/qF,uC,IAA+qF,IAA/qF,gB,EAA+qF,QAA/qF,6B,EAAysF,UAAzsF,uC,IAAysF,IAAzsF,gB,EAAysF,QAAzsF,6B,EAAmtF,UAAntF,uC,IAAmtF,IAAntF,gB,EAAmtF,QAAntF,6B,EAA6tF,UAA7tF,uC,IAA6tF,IAA7tF,gB,EAA6tF,QAA7tF,6B,EAAuvF,UAAvvF,uC,IAAuvF,IAAvvF,gB,EAAuvF,QAAvvF,6B,EAAiwF,UAAjwF,uC,IAAiwF,IAAjwF,gB,EAAiwF,QAAjwF,6B,EAA2wF,UAA3wF,uC,IAA2wF,IAA3wF,gB,EAA2wF,QAA3wF,6B,EAAqxF,UAArxF,uC,IAAqxF,IAArxF,gB,EAAqxF,QAArxF,6B,EAA+xF,UAA/xF,uC,IAA+xF,IAA/xF,gB,EAA+xF,QAA/xF,6B,EAAyzF,UAAzzF,uC,IAAyzF,IAAzzF,gB,EAAyzF,QAAzzF,6B,EAAm0F,UAAn0F,uC,IAAm0F,IAAn0F,gB,EAAm0F,QAAn0F,6B,EAA60F,UAA70F,uC,IAA60F,IAA70F,gB,EAA60F,QAA70F,6B,EAAu1F,UAAv1F,uC,IAAu1F,IAAv1F,gB,EAAu1F,QAAv1F,6B,EAAi2F,UAAj2F,uC,IAAi2F,IAAj2F,gB,EAAi2F,QAAj2F,6B,EAA22F,UAA32F,uC,IAA22F,IAA32F,gB,EAA22F,QAA32F,6B,EAAq3F,UAAr3F,uC,IAAq3F,IAAr3F,gB,EAAq3F,QAAr3F,6B,EAA+3F,UAA/3F,uC,IAA+3F,IAA/3F,gB,EAA+3F,QAA/3F,6B,EAAy4F,UAAz4F,uC,IAAy4F,IAAz4F,gB,EAAy4F,QAAz4F,6B,EAAm5F,UAAn5F,uC,IAAm5F,IAAn5F,gB,EAAm5F,QAAn5F,6B,EAA65F,UAA75F,uC,IAA65F,IAA75F,gB,EAA65F,QAA75F,6B,EAAu6F,UAAv6F,uC,IAAu6F,IAAv6F,gB,EAAu6F,QAAv6F,6B,EAAi7F,UAAj7F,uC,IAAi7F,IAAj7F,gB,EAAi7F,QAAj7F,6B,EAA27F,UAA37F,uC,IAA27F,IAA37F,gB,EAA27F,QAA37F,6B,EAAq8F,UAAr8F,uC,IAAq8F,IAAr8F,gB,EAAq8F,QAAr8F,6B,EAA+8F,UAA/8F,uC,IAA+8F,IAA/8F,gB,EAA+8F,QAA/8F,6B,EAAy9F,UAAz9F,uC,IAAy9F,IAAz9F,gB,EAAy9F,QAAz9F,6B,EAAm+F,UAAn+F,uC,IAAm+F,IAAn+F,gB,EAAm+F,QAAn+F,6B,EAA6+F,UAA7+F,uC,IAA6+F,IAA7+F,gB,EAA6+F,QAA7+F,6B,EAAuhG,UAAvhG,uC,IAAuhG,IAAvhG,gB,EAAuhG,QAAvhG,6B,EAAiiG,UAAjiG,uC,IAAiiG,IAAjiG,gB,EAAiiG,QAAjiG,6B,EAA2iG,UAA3iG,uC,IAA2iG,IAA3iG,gB,EAA2iG,QAA3iG,6B,EAAqjG,UAArjG,uC,IAAqjG,IAArjG,gB,EAAqjG,QAArjG,6B,EAA+jG,UAA/jG,uC,IAA+jG,IAA/jG,gB,EAA+jG,QAA/jG,6B,EAAylG,UAAzlG,uC,IAAylG,IAAzlG,gB,EAAylG,QAAzlG,6B,EAAmmG,UAAnmG,uC,IAAmmG,IAAnmG,gB,EAAmmG,QAAnmG,6B,EAA6mG,UAA7mG,uC,IAA6mG,IAA7mG,gB,EAA6mG,QAA7mG,6B,EAAuoG,UAAvoG,uC,IAAuoG,IAAvoG,gB,EAAuoG,QAAvoG,6B,EAAipG,UAAjpG,uC,IAAipG,IAAjpG,gB,EAAipG,QAAjpG,6B,EAA2pG,UAA3pG,uC,IAA2pG,IAA3pG,gB,EAA2pG,QAA3pG,6B,EAAqqG,UAArqG,uC,IAAqqG,IAArqG,gB,EAAqqG,QAArqG,6B,EAA+qG,UAA/qG,uC,IAA+qG,IAA/qG,gB,EAA+qG,QAA/qG,6B,EAAysG,UAAzsG,uC,IAAysG,IAAzsG,gB,EAAysG,QAAzsG,6B,EAAmtG,UAAntG,uC,IAAmtG,IAAntG,gB,EAAmtG,QAAntG,6B,EAA6tG,UAA7tG,uC,IAA6tG,IAA7tG,gB,EAA6tG,QAA7tG,6B,EAAuvG,UAAvvG,uC,IAAuvG,IAAvvG,gB,EAAuvG,QAAvvG,6B,EAAiwG,UAAjwG,uC,IAAiwG,IAAjwG,gB,EAAiwG,QAAjwG,6B,EAA2wG,UAA3wG,uC,IAA2wG,IAA3wG,gB,EAA2wG,QAA3wG,6B,EAAqxG,UAArxG,uC,IAAqxG,IAArxG,gB,EAAqxG,QAArxG,6B,EAA+xG,UAA/xG,uC,IAA+xG,IAA/xG,gB,EAA+xG,QAA/xG,6B,EAAyzG,UAAzzG,uC,IAAyzG,IAAzzG,gB,EAAyzG,QAAzzG,6B,EAAm0G,UAAn0G,uC,IAAm0G,IAAn0G,gB,EAAm0G,QAAn0G,6B,EAA60G,UAA70G,uC,IAA60G,IAA70G,gB,EAA60G,QAA70G,6B,EAAu1G,UAAv1G,uC,IAAu1G,IAAv1G,gB,EAAu1G,QAAv1G,6B,EAAi2G,UAAj2G,uC,IAAi2G,IAAj2G,gB,EAAi2G,QAAj2G,6B,EAA22G,UAA32G,uC,IAA22G,IAA32G,gB,EAA22G,QAA32G,6B,EAAq3G,UAAr3G,uC,IAAq3G,IAAr3G,gB,EAAq3G,QAAr3G,6B,EAA+3G,UAA/3G,uC,IAA+3G,IAA/3G,gB,EAA+3G,QAA/3G,6B,EAAy4G,UAAz4G,uC,IAAy4G,IAAz4G,gB,EAAy4G,QAAz4G,6B,EAAm5G,UAAn5G,uC,IAAm5G,IAAn5G,gB,EAAm5G,QAAn5G,6B,EAA65G,UAA75G,uC,IAA65G,IAA75G,gB,EAA65G,QAA75G,6B,EAAu6G,UAAv6G,uC,IAAu6G,IAAv6G,gB,EAAu6G,QAAv6G,6B,EAAi7G,UAAj7G,uC,IAAi7G,IAAj7G,gB,EAAi7G,QAAj7G,6B,EAA27G,UAA37G,uC,IAA27G,IAA37G,gB,EAA27G,QAA37G,6B,EAAq8G,UAAr8G,uC,IAAq8G,IAAr8G,gB,EAAq8G,QAAr8G,6B,EAA+8G,UAA/8G,uC,IAA+8G,IAA/8G,gB,EAA+8G,QAA/8G,6B,EAAy9G,UAAz9G,uC,IAAy9G,IAAz9G,gB,EAAy9G,QAAz9G,6B,EAAm+G,UAAn+G,uC,IAAm+G,IAAn+G,gB,EAAm+G,QAAn+G,6B,EAA6+G,UAA7+G,uC,IAA6+G,IAA7+G,gB,EAA6+G,QAA7+G,6B,EAAuhH,UAAvhH,uC,IAAuhH,IAAvhH,gB,EAAuhH,QAAvhH,6B,EAAiiH,UAAjiH,uC,IAAiiH,IAAjiH,gB,EAAiiH,QAAjiH,6B,EAA2iH,UAA3iH,uC,IAA2iH,IAA3iH,gB,EAA2iH,QAA3iH,6B,EAAqjH,UAArjH,uC,IAAqjH,IAArjH,gB,EAAqjH,QAArjH,6B,EAA+jH,UAA/jH,uC,IAA+jH,IAA/jH,gB,EAA+jH,QAA/jH,6B,EAAylH,UAAzlH,uC,IAAylH,IAAzlH,gB,EAAylH,QAAzlH,6B,EAAmmH,UAAnmH,uC,IAAmmH,IAAnmH,gB,EAAmmH,QAAnmH,6B,EAA6mH,UAA7mH,uC,IAA6mH,IAA7mH,gB,EAA6mH,QAA7mH,6B,EAAuoH,UAAvoH,uC,IAAuoH,IAAvoH,gB,EAAuoH,QAAvoH,6B,EAAipH,UAAjpH,uC,IAAipH,IAAjpH,gB,EAAipH,QAAjpH,6B,EAA2pH,UAA3pH,uC,IAA2pH,IAA3pH,gB,EAA2pH,QAA3pH,6B,EAAqqH,UAArqH,uC,IAAqqH,IAArqH,gB,EAAqqH,QAArqH,6B,EAA+qH,UAA/qH,uC,IAA+qH,IAA/qH,gB,EAA+qH,QAA/qH,6B,EAAysH,UAAzsH,uC,IAAysH,IAAzsH,gB,EAAysH,QAAzsH,6B,EAAmtH,UAAntH,uC,IAAmtH,IAAntH,gB,EAAmtH,QAAntH,6B,EAA6tH,UAA7tH,uC,IAA6tH,IAA7tH,gB,EAA6tH,QAA7tH,6B,EAAuvH,UAAvvH,uC,IAAuvH,IAAvvH,gB,EAAuvH,QAAvvH,6B,EAAiwH,UAAjwH,uC,IAAiwH,IAAjwH,gB,EAAiwH,QAAjwH,6B,EAA2wH,UAA3wH,uC,IAA2wH,IAA3wH,gB,EAA2wH,QAA3wH,6B,EAAqxH,UAArxH,uC,IAAqxH,IAArxH,gB,EAAqxH,QAArxH,6B,EAA+xH,UAA/xH,uC,IAA+xH,IAA/xH,gB,EAA+xH,QA

EkOtIvC,ClOsIuC,C;MkOrIpD,OlOmDkE,YkOnDvD,ElOmDwE,KAAjB,EkOnDjD,ElOmD8E,KAA7B,CkOnDv
D,KAAX,GlOkFsD,SkOlFjC,ElOkF2C,KAAK,GkOlF3C,ElOkFuD,KAAZ,IAAf,CkOlFtD,GlOqEqD,SAAU,CAaT
,SkOlFpB,ElOkF8B,KAAK,GkOlF9B,ElOkF0C,KAAZ,IAAf,CAbS,MAAK,GkOrExB,ClOqEmC,KAAX,IAAf,C;
K;IkOlEzD,qC;MACI,SlNwIsD,ekNxI7C,ClNwI6C,EkNxIzC,ClNwIyC,C;MkNvItD,SlNuIsD,ekNvI7C,ClNuI6C,E
kNvIzC,ClNuIyC,C;MkNtItD,OlNqDmE,akNrDxD,ElNqD0E,KAAlB,EkNrDlD,ElNqDgF,KAA9B,CkNrDxD,KA
AX,GlN+EwD,UkN/EnC,ElN+E8C,KAAK,UkN/E9C,ElN+E0D,KAAZ,CAAhB,CkN/ExD,GlNkEuD,UAAW,CAa
V,UkN/EtB,ElN+EiC,KAAK,UkN/EjC,ElN+E6C,KAAZ,CAAhB,CAbU,MAAK,KkNlE3B,ClNkEsC,KAAX,CAA
hB,C;K;IkN/D3D,uD;MAmBI,WAAO,CAAP,C;QAD8E,OlOwBZ,YkOvBlD,KlOuBmE,KAAjB,EkOvBzC,GlOuB
sE,KAA7B,CkOvBlD,KAD8D,GAChD,GADgD,GlOuDxB,SkOtDf,GlOsDyB,KAAK,GkOtDxB,mBAAiB,GAAjB
,EAAsB,KAAtB,ElO2WV,SkO3WuC,IlO2WvC,CkO3WU,ClOsDoC,KAAZ,IAAf,C;akOrDtD,WAAO,CAAP,C;Q
AF8E,OlOwBZ,YkOtBlD,KlOsBmE,KAAjB,EkOtBzC,GlOsBsE,KAA7B,CkOtBlD,KAF8D,GAEhD,GAFgD,GlO
0CzB,SkOxCd,GlOwCwB,KAAK,GkOxCvB,mBAAiB,KAAjB,EAAwB,GAAxB,ElO0WV,SkO1WwC,CAAC,IA
AD,IlO0WxC,CkO1WU,ClOwCkC,KAAX,IAAf,C;;QkOvC7C,MAAa,gCAAyB,eAAzB,C;K;IAGzB,uD;MAmBI,
sBAAO,CAAP,C;QADkF,OlNQf,akNPnD,KlNOqE,KAAlB,EkNP1C,GlNOwE,KAA9B,CkNPnD,KADkE,GACp
D,GADoD,GlNkC1B,UkNjCjB,GlNiC4B,KAAK,UkNjC3B,mBAAiB,GAAjB,EAAsB,KAAtB,ElNkWP,UkNlWo
C,IlNkWpC,CkNlWO,ClNiCuC,KAAZ,CAAhB,C;akNhCxD,sBAAO,CAAP,C;QAFkF,OlNQf,akNNnD,KlNMqE,
KAAlB,EkNN1C,GlNMwE,KAA9B,CkNNnD,KAFkE,GAEpD,GAFoD,GlNqB3B,UkNnBhB,GlNmB2B,KAAK,
KkNnB1B,mBAAiB,KAAjB,EAAwB,GAAxB,ElNiWP,UkNjWsC,IAAD,alNiWrC,CkNjWO,ClNmBqC,KAAX,C
AAhB,C;;QkNlB/C,MAAa,gCAAyB,eAAzB,C;K;IjOlDC,sB;MAEtB,8B;MAFyD,gB;K;IAEzD,4B;MAAA,gC;M
ACI,iBAGqC,WAAO,CAAP,C;MAErC,iBAGqC,WAAO,CAAP,C;MAErC,kBAGmC,C;MAEnC,iBAGkC,E;K;;;I
AnBtC,wC;MAAA,uC;QAAA,sB;;MAAA,gC;K;wGAsBA,iB;MAM0D,OAAa,0BA6OjC,SAAL,GAAiB,KA7OqB
,EAAU,KF4O3C,KAAL,GAAiB,GE5OqB,C;K;oGAEvE,iB;MAOoE,OAAa,0BAoO3C,SAAL,GAAiB,KApO+B,E
AAU,KAoOrD,KAAL,GAAiB,KApO+B,C;K;wGAEjF,yB;MA2PA,6B;MD5PA,8C;MCCA,wB;QAMyD,ODAS,Y
AAiB,CC8PhD,cAAU,SAAL,GAAiB,KAAtB,CD9PgD,MAAjB,ECAe,KDAc,KAA7B,C;O;KCNlE,C;wGAQA,yB
;MA6PA,aAS6D,0B;MAT7D,+B;Me9PA,gD;MfCA,wB;QAM0D,OeAS,aAAkB,CfgQhD,eAAW,oBAAL,SAAK,
CAAL,YAAN,CehQgD,MAAlB,EfAgB,KeAc,KAA9B,C;O;KfNnE,C;8FAQA,yB;MA2QA,2B;MA3QA,wB;QAE
sD,ODMD,cAAU,CC4O5B,cAAU,SAAL,GAAiB,KAAtB,CD5O4B,MAAK,GAAW,CD2O5C,cEjPsC,KFiP5B,K
AAL,GAAiB,GAAtB,CC3O4C,MAAX,IAAf,C;O;KCRrD,C;8FAGA,yB;MAwQA,2B;MAxQA,wB;QAEuD,ODG
F,cAAU,CC4O5B,cAAU,SAAL,GAAiB,KAAtB,CD5O4B,MAAK,GAAW,CC4O5C,cA/OuC,KA+O7B,KAAL,G
AAiB,KAAtB,CD5O4C,MAAX,IAAf,C;O;KCLrD,C;8FAGA,yB;MAqQA,2B;MArQA,wB;QAEqD,ODAA,cAAU
,CC4O5B,cAAU,SAAL,GAAiB,KAAtB,CD5O4B,MAAK,GCAI,KDAO,KAAX,IAAf,C;O;KCFrD,C;8FAGA,yB;
MA4QA,aAS6D,0B;MAT7D,+B;MA5OA,wB;QAEuD,OeAA,eAAW,CfmP7B,eAAW,oBAAL,SAAK,CAAL,YA
AN,CenP6B,MAAK,KfAI,KeAO,KAAX,CAAhB,C;O;KfFvD,C;gGAIA,yB;MA8QA,aAS8D,0B;MAT9D,+B;MA
8NA,6B;MA9NA,wB;QAEuD,ODMD,cAAU,CC+N7B,cAAU,SAAL,GAAiB,KAAtB,CD/N6B,MAAK,GAAY,CD8N9C,cEpOwC,KFoO9B,KAA
L,GAAiB,GAAtB,CC9N8C,MAAZ,IAAf,C;O;KCRtD,C;gGAIA,yB;MAmRA,aAS+D,0B;MAT/D,+B;MA2NA,6B;MA3NA,wB;QAEwD,ODGF,c
AAU,CC+N7B,cAAU,SAAL,GAAiB,KAAtB,CD/N6B,MAAK,GAAY,CC+N9C,cAlOyC,KAkO/B,KAAL,GAAiB
,KAAtB,CD/N8C,MAAZ,IAAf,C;O;KCLtD,C;gGAIA,yB;MAwRA,aAS+D,0B;MAT/D,+B;MAwNA,6B;MAxNA,wB;QAEsD,ODAA,cAAU,CC+N7B,cAAU,SAAL,GAAiB,KAAtB,CD/N6B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;MA6RA,aAS+D,0B;MAT/D,+B;MAqNA,6B;MArNA,wB;QAEwD,OeAA,eAAW,CfsO9B,eAAW,oBAAL,SAAK,CAAL,YAAN,CetO8B,MAAK,KfAK,KeAO,KAAZ,CAAhB,C;O;KfFxD,C;gGAIA,yB;MAkSA,aAS+D,0B;MAT/D,+B;MAiNA,6B;MAjNA,wB;QAEwD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAuSA,aAS+D,0B;MAT/D,+B;MA+MA,6B;MA/MA,wB;QAEyD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CCA/C,cAvCyC,KAuC/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MA4SA,aAS+D,0B;MAT/D,+B;MAkMA,6B;MAlMA,wB;QAEuD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GCAM,KDAO,KAAb,IAAf,C;O;KCFvD,C;gGAIA,yB;
MAiTA,aAS+D,0B;MAT/D,+B;MAyLA,6B;MAzLA,wB;QAEyD,OeAA,eAAW,CfyO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CezO8B,MAAK,KfAM,KeAO,KAAb,CAAhB,C;O;KfFzD,C;gGAIA,yB;MAsTA,aAS+D,0B;MAT/D,+B;MAwLA,6B;MAxLA,wB;QAEyD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MA2TA,aAS+D,0B;MAT/D,+B;MAsLA,6B;MAtLA,wB;QAEwD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CCA/C,cAjDyC,KAiD/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAgUA,aAS+D,0B;MAT/D,+B;MA4KA,6B;MA5KA,wB;QAEsD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;
MAqUA,aAS+D,0B;MAT/D,+B;MAmKA,6B;MAnKA,wB;QAEwD,OeAA,eAAW,CfkO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CelO8B,MAAK,KfAK,KeAO,KAAZ,CAAhB,C;O;KfFxD,C;gGAIA,yB;MA0UA,aAS+D,0B;MAT/D,+B;MAkKA,6B;MAlKA,wB;QAEwD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MA+UA,aAS+D,0B;MAT/D,+B;MAgKA,6B;MAhKA,wB;QAEyD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CCA/C,cA3DyC,KA2D/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAoVA,aAS+D,0B;MAT/D,+B;MA8JA,6B;MA9JA,wB;QAEuD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GCAM,KDAO,KAAb,IAAf,C;O;KCFvD,C;gGAIA,yB;
MAyVA,aAS+D,0B;MAT/D,+B;MA6JA,6B;MA7JA,wB;QAEyD,OeAA,eAAW,CfuO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CevO8B,MAAK,KfAM,KeAO,KAAb,CAAhB,C;O;KfFzD,C;gGAIA,yB;MA8VA,aAS+D,0B;MAT/D,+B;MA4JA,6B;MA5JA,wB;QAEyD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAmWA,aAS+D,0B;MAT/D,+B;MA0JA,6B;MA1JA,wB;QAEwD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CCA/C,cAvEyC,KAuE/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAwWA,aAS+D,0B;MAT/D,+B;MAoJA,6B;MApJA,wB;QAEsD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;
MA6WA,aAS+D,0B;MAT/D,+B;MAyJA,6B;MAzJA,wB;QAEwD,OeAA,eAAW,CfwO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CexO8B,MAAK,KfAK,KeAO,KAAZ,CAAhB,C;O;KfFxD,C;gGAIA,yB;MAkXA,aAS+D,0B;MAT/D,+B;MAwJA,6B;MAxJA,wB;QAEwD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAuXA,aAS+D,0B;MAT/D,+B;MAsJA,6B;MAtJA,wB;QAEyD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CCA/C,cAnFyC,KAmF/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MA4XA,aAS+D,0B;MAT/D,+B;MAkJA,6B;MAlJA,wB;QAEuD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GCAM,KDAO,KAAb,IAAf,C;O;KCFvD,C;gGAIA,yB;
MAiYA,aAS+D,0B;MAT/D,+B;MAqJA,6B;MArJA,wB;QAEyD,OeAA,eAAW,CfyO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CezO8B,MAAK,KfAM,KeAO,KAAb,CAAhB,C;O;KfFzD,C;gGAIA,yB;MAsYA,aAS+D,0B;MAT/D,+B;MAoJA,6B;MApJA,wB;QAEyD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MA2YA,aAS+D,0B;MAT/D,+B;MAkJA,6B;MAlJA,wB;QAEwD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CCA/C,cAvFyC,KAuF/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAgZA,aAS+D,0B;MAT/D,+B;MA8IA,6B;MA9IA,wB;QAEsD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;
MAqZA,aAS+D,0B;MAT/D,+B;MAiJA,6B;MAjJA,wB;QAEwD,OeAA,eAAW,CfoO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CepO8B,MAAK,KfAK,KeAO,KAAZ,CAAhB,C;O;KfFxD,C;gGAIA,yB;MA0ZA,aAS+D,0B;MAT/D,+B;MA8IA,6B;MA9IA,wB;QAEwD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MA+ZA,aAS+D,0B;MAT/D,+B;MA4IA,6B;MA5IA,wB;QAEyD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CCA/C,cAnGyC,KAmG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAobA,aAS+D,0B;MAT/D,+B;MA0IA,6B;MA1IA,wB;QAEuD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GCAM,KDAO,KAAb,IAAf,C;O;KCFvD,C;gGAIA,yB;
MAybA,aAS+D,0B;MAT/D,+B;MAyIA,6B;MAzIA,wB;QAEyD,OeAA,eAAW,CfqO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CerO8B,MAAK,KfAM,KeAO,KAAb,CAAhB,C;O;KfFzD,C;gGAIA,yB;MAscA,aAS+D,0B;MAT/D,+B;MAwIA,6B;MAxIA,wB;QAEyD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAmcA,aAS+D,0B;MAT/D,+B;MAsIA,6B;MAtIA,wB;QAEwD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAzGyC,KAyG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAwcA,aAS+D,0B;MAT/D,+B;MAkIA,6B;MAlIA,wB;QAEsD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;
MA6cA,aAS+D,0B;MAT/D,+B;MAiIA,6B;MAjIA,wB;QAEwD,OeAA,eAAW,CfsO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CetO8B,MAAK,KfAK,KeAO,KAAZ,CAAhB,C;O;KfFxD,C;gGAIA,yB;MAkdA,aAS+D,0B;MAT/D,+B;MAkIA,6B;MAlIA,wB;QAEwD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAudA,aAS+D,0B;MAT/D,+B;MAgIA,6B;MAhIA,wB;QAEyD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAvGyC,KAuG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAodA,aAS+D,0B;MAT/D,+B;MA4HA,6B;MA5HA,wB;QAEuD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAAtB,CDA8B,MAAK,GCAM,KDAO,KAAb,IAAf,C;O;KCFvD,C;gGAIA,yB;
MAqdA,aAS+D,0B;MAT/D,+B;MAuHA,6B;MAvHA,wB;QAEyD,OeAA,eAAW,CfmO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CenO8B,MAAK,KfAM,KeAO,KAAb,CAAhB,C;O;KfFzD,C;gGAIA,yB;MAsdA,aAS+D,0B;MAT/D,+B;MAsHA,6B;MAtHA,wB;QAEyD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAmdA,aAS+D,0B;MAT/D,+B;MAoHA,6B;MApHA,wB;QAEwD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAjGyC,KAiG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAwdA,aAS+D,0B;MAT/D,+B;MAkHA,6B;MAlHA,wB;QAEsD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;
MA6dA,aAS+D,0B;MAT/D,+B;MAiHA,6B;MAjHA,wB;QAEwD,OeAA,eAAW,CfoO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CepO8B,MAAK,KfAK,KeAO,KAAZ,CAAhB,C;O;KfFxD,C;gGAIA,yB;MAkeA,aAS+D,0B;MAT/D,+B;MAkHA,6B;MAlHA,wB;QAEwD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAueA,aAS+D,0B;MAT/D,+B;MAgHA,6B;MAhHA,wB;QAEyD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAvGyC,KAuG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAoeA,aAS+D,0B;MAT/D,+B;MA4GA,6B;MA5GA,wB;QAEuD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAM,KDAO,KAAb,IAAf,C;O;KCFvD,C;gGAIA,yB;
MAqeA,aAS+D,0B;MAT/D,+B;MAuGA,6B;MAvGA,wB;QAEyD,OeAA,eAAW,CfmO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CenO8B,MAAK,KfAM,KeAO,KAAb,CAAhB,C;O;KfFzD,C;gGAIA,yB;MAseA,aAS+D,0B;MAT/D,+B;MAsGA,6B;MAtGA,wB;QAEyD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAmeA,aAS+D,0B;MAT/D,+B;MAoGA,6B;MApGA,wB;QAEwD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAjGyC,KAiG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAweA,aAS+D,0B;MAT/D,+B;MAkGA,6B;MAlGA,wB;QAEsD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;
MA6eA,aAS+D,0B;MAT/D,+B;MAiGA,6B;MAjGA,wB;QAEwD,OeAA,eAAW,CfoO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CepO8B,MAAK,KfAK,KeAO,KAAZ,CAAhB,C;O;KfFxD,C;gGAIA,yB;MAkfA,aAS+D,0B;MAT/D,+B;MAkGA,6B;MAlGA,wB;QAEwD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MA+fA,aAS+D,0B;MAT/D,+B;MAgGA,6B;MAhGA,wB;QAEyD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAvGyC,KAuG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAogBA,aAS+D,0B;MAT/D,+B;MA4FA,6B;MA5FA,wB;QAEuD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAM,KDAO,KAAb,IAAf,C;O;KCFvD,C;gGAIA,yB;
MAqgBA,aAS+D,0B;MAT/D,+B;MAuFA,6B;MAvFA,wB;QAEyD,OeAA,eAAW,CfmO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CenO8B,MAAK,KfAM,KeAO,KAAb,CAAhB,C;O;KfFzD,C;gGAIA,yB;MAshBA,aAS+D,0B;MAT/D,+B;MAsFA,6B;MAtFA,wB;QAEyD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAmhBA,aAS+D,0B;MAT/D,+B;MAoFA,6B;MApFA,wB;QAEwD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAjGyC,KAiG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAwhBA,aAS+D,0B;MAT/D,+B;MAkFA,6B;MAlFA,wB;QAEsD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;
MA6hBA,aAS+D,0B;MAT/D,+B;MAiFA,6B;MAjFA,wB;QAEwD,OeAA,eAAW,CfoO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CepO8B,MAAK,KfAK,KeAO,KAAZ,CAAhB,C;O;KfFxD,C;gGAIA,yB;MAkiBA,aAS+D,0B;MAT/D,+B;MAkFA,6B;MAlFA,wB;QAEwD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MA+iBA,aAS+D,0B;MAT/D,+B;MAgFA,6B;MAhFA,wB;QAEyD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAvGyC,KAuG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAojBA,aAS+D,0B;MAT/D,+B;MA4EA,6B;MA5EA,wB;QAEuD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAM,KDAO,KAAb,IAAf,C;O;KCFvD,C;gGAIA,yB;
MAqjBA,aAS+D,0B;MAT/D,+B;MAuEA,6B;MAvEA,wB;QAEyD,OeAA,eAAW,CfmO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CenO8B,MAAK,KfAM,KeAO,KAAb,CAAhB,C;O;KfFzD,C;gGAIA,yB;MAskBA,aAS+D,0B;MAT/D,+B;MAsEA,6B;MAtEA,wB;QAEyD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAmkBA,aAS+D,0B;MAT/D,+B;MAoEA,6B;MApEA,wB;QAEwD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAjGyC,KAiG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAwkBA,aAS+D,0B;MAT/D,+B;MAkEA,6B;MAlEA,wB;QAEsD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;
MA6kBA,aAS+D,0B;MAT/D,+B;MAiEA,6B;MAjEA,wB;QAEwD,OeAA,eAAW,CfoO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CepO8B,MAAK,KfAK,KeAO,KAAZ,CAAhB,C;O;KfFxD,C;gGAIA,yB;MAklBA,aAS+D,0B;MAT/D,+B;MAkEA,6B;MAlEA,wB;QAEwD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MA+lBA,aAS+D,0B;MAT/D,+B;MAgEA,6B;MAhEA,wB;QAEyD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAvGyC,KAuG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAomBA,aAS+D,0B;MAT/D,+B;MA4DA,6B;MA5DA,wB;QAEuD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAM,KDAO,KAAb,IAAf,C;O;KCFvD,C;gGAIA,yB;
MAqmBA,aAS+D,0B;MAT/D,+B;MAuDA,6B;MAvDA,wB;QAEyD,OeAA,eAAW,CfmO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CenO8B,MAAK,KfAM,KeAO,KAAb,CAAhB,C;O;KfFzD,C;gGAIA,yB;MAsmBA,aAS+D,0B;MAT/D,+B;MAsDA,6B;MAtDA,wB;QAEyD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAmnBA,aAS+D,0B;MAT/D,+B;MAoDA,6B;MApDA,wB;QAEwD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAjGyC,KAiG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAwnBA,aAS+D,0B;MAT/D,+B;MAkDA,6B;MAlDA,wB;QAEsD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;
MA6nBA,aAS+D,0B;MAT/D,+B;MAiDA,6B;MAjDA,wB;QAEwD,OeAA,eAAW,CfoO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CepO8B,MAAK,KfAK,KeAO,KAAZ,CAAhB,C;O;KfFxD,C;gGAIA,yB;MAkoBA,aAS+D,0B;MAT/D,+B;MAkDA,6B;MAlDA,wB;QAEwD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MA+oBA,aAS+D,0B;MAT/D,+B;MAgDA,6B;MAhDA,wB;QAEyD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAvGyC,KAuG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAopBA,aAS+D,0B;MAT/D,+B;MA4CA,6B;MA5CA,wB;QAEuD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAM,KDAO,KAAb,IAAf,C;O;KCFvD,C;gGAIA,yB;
MAqpBA,aAS+D,0B;MAT/D,+B;MAuCA,6B;MAvCA,wB;QAEyD,OeAA,eAAW,CfmO9B,eAAW,oBAAL,SAAK,
CAAL,YAAN,CenO8B,MAAK,KfAM,KeAO,KAAb,CAAhB,C;O;KfFzD,C;gGAIA,yB;MAsqBA,aAS+D,0B;MAT/D,+B;MAsCA,6B;MAtCA,wB;QAEyD,ODMD,cAAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CDC/C,cEDsC,KFCjC,KAAL,
GAAiB,GAtB,CCA+C,MAAb,IAAf,C;O;KCRvD,C;gGAIA,yB;MAmrBA,aAS+D,0B;MAT/D,+B;MAoCA,6B;MApCA,wB;QAEwD,ODGF,c
AAU,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GAAa,CCA/C,cAjGyC,KAiG/B,KAAL,GAAiB
,KAAtB,CDA+C,MAAb,IAAf,C;O;KCLvD,C;gGAIA,yB;MAwrBA,aAS+D,0B;MAT/D,+B;MAkCA,6B;MAlCA,wB;QAEsD,ODAA,cAAU
,CCAe,cAAU,SAAL,GAAiB,KAtB,CDA8B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;O;KCFtD,C;gGAIA,yB;
MA6rBA,aAS+D,0B;MAT/D,+B;MAiCA,6B;MAjCA,wB;QAEwD,OeAA,eA

GAAiB,GAAtB,CCpMiB,C;O;KCRpD,C;4FAGA,yB;MAiMA,6B;MD9LA,4C;MCHA,wB;QAEsD,ODGF,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECqMjB,cAxMqC,KAwM3B,KAAL,GAAiB,KAAtB,CDrMiB,C;O;KCLpD,C;4FAGA,yB;MA8LA,6B;MD9LA,4C;MCAA,wB;QAEoD,ODAA,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCFpD,C;4FAGA,yB;MAqMA,aAS6D,0B;MAT7D,+B;MerMA,8C;MfAA,wB;QAEsD,OeAA,Yf4MjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce5MiB,EfAmB,KeAnB,C;O;KfFtD,C;4FAIA,yB;MAuLA,6B;MDzKA,kD;MCdA,wB;QAMqD,ODcD,cC4KjB,cAAU,SAAL,GAAiB,KAAtB,CD5KiB,ED2KjB,cEzLoC,KFyL1B,KAAL,GAAiB,GAAtB,CC3KiB,C;O;KCpBpD,C;4FAOA,yB;MAgLA,6B;MDzKA,kD;MCPA,wB;QAMsD,ODOF,cC4KjB,cAAU,SAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCbpD,C;4FAOA,yB;MAyKA,6B;MDzKA,kD;MCAA,wB;QAMoD,ODAA,cC4KjB,cAAU,SAAL,GAAiB,KAAtB,CD5KiB,ECAkB,KDAlB,C;O;KCNpD,C;4FAOA,yB;MA4KA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfNtD,C;sGAQA,yB;MA0JA,6B;MD9LA,4C;MCoCA,wB;QAMiD,ODxCG,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,EDoMjB,cE5JqC,KF4J3B,KAAL,GAAiB,GAAtB,CC3KiB,C;O;KCkCpD,C;sGAOA,yB;MAmJA,6B;MD9LA,4C;MC2CA,wB;QAMkD,OD/C/CE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECqMjB,cAtJsC,KAsJ5B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCyCpD,C;sGAOA,yB;MA4IA,6B;MD9LA,4C;MCkDA,wB;QAMmD,ODtDlDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECsDmB,KDtDnB,C;O;KCgDpD,C;sGAOA,yB;MAiIA,aAS6D,0B;MAT7D,+B;MerMA,8C;MfsDA,wB;QAMoD,OetDnDE,Yf4MjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce5MiB,Ef0DoB,Ke1DpB,C;O;KfoDtD,C;4FAIA,yB;MA6HA,6B;MDzKA,kD;MCuDA,wB;QAMqD,ODvDpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cE7HqC,KF6H3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCwDpD,C;4FAOA,yB;MAsHA,6B;MDzKA,kD;MC4DA,wB;QAMoD,OD5DnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC6DpD,C;4FAOA,yB;MA2GA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkEtD,C;4FAOA,yB;MAqGA,6B;MDzKA,kD;MCiEA,wB;QAMqD,ODjEpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEvGqC,KFuG3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCkEpD,C;4FAOA,yB;MAgGA,6B;MDzKA,kD;MCsEA,wB;QAMoD,ODtEnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCuEpD,C;4FAOA,yB;MAqFA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwEtD,C;4FAOA,yB;MA0EA,6B;MDzKA,kD;MC2EA,wB;QAMqD,OD3EpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEjEqC,KFiE3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC4EpD,C;4FAOA,yB;MAuEA,6B;MDzKA,kD;MCgFA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiFpD,C;4FAOA,yB;MA+DA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkFtD,C;4FAOA,yB;MAoDA,6B;MDzKA,kD;MCqFA,wB;QAMqD,ODrFpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cE3DqC,KF2D3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCsFpD,C;4FAOA,yB;MAiDA,6B;MDzKA,kD;MC0FA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC2FpD,C;4FAOA,yB;MAyCA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwFtD,C;4FAOA,yB;MA8BA,6B;MDzKA,kD;MC+FA,wB;QAMqD,OD/FpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cErBqC,KFqB3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCgGpD,C;4FAOA,yB;MA2BA,6B;MDzKA,kD;MCoGA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCqGpD,C;4FAOA,yB;MAmBA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkGtD,C;4FAOA,yB;MAgBA,6B;MDzKA,kD;MCyGA,wB;QAMqD,ODzGpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEPqC,KFO3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC0GpD,C;4FAOA,yB;MAaA,6B;MDzKA,kD;MC8GA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC+GpD,C;4FAOA,yB;MAUA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwHtD,C;4FAOA,yB;MAOA,6B;MDzKA,kD;MCmHA,wB;QAMqD,ODnHpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEdqC,KFc3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCoHpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCwHA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCyHpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkItD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC6HA,wB;QAMqD,OD7HpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC8HpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCkIA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCmIpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwItD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCuIA,wB;QAMqD,ODvIpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCwIpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC4IA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC6IpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkJtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCiJA,wB;QAMqD,ODjJpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCkJpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCsJA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCuJpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwJtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC2JA,wB;QAMqD,OD3JpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC4JpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCgKA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiKpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkKtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCqKA,wB;QAMqD,ODrKpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCsKpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC0KA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC2KpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfoLtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCyLA,wB;QAMqD,ODzLpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC0LpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC8LA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiMpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwMtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCmMA,wB;QAMqD,ODnMpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCoMpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCwMA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCyMpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkNtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC6MA,wB;QAMqD,OD7MpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC8MpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCkNA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCmNpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwNtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCuNA,wB;QAMqD,ODvNpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCwNpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC4NA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC6NpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkOtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCiOA,wB;QAMqD,ODjOpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCkOpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCsOA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCuOpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwOtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC2OA,wB;QAMqD,OD3OpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC4OpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCgPA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiPpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkPtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCqPA,wB;QAMqD,ODrPpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCsPpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC0PA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC2PpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwPtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCyPA,wB;QAMqD,OzPzPpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC0PpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC8PA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiQpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkQtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCqQA,wB;QAMqD,ODrQpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCsQpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC0QA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC2QpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwQtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC2QA,wB;QAMqD,OzPzPpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC4QpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCgRA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiRpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkRtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCqRA,wB;QAMqD,ODrRpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCsRpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC0RA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC2RpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwRtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC2RA,wB;QAMqD,OzPzPpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC4RpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCgSA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiSpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkStD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCqSA,wB;QAMqD,ODrSpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCsSpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC0SA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC2SpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwStD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC2SA,wB;QAMqD,OzPzPpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC4SpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCgTA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiTpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkTtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCqTA,wB;QAMqD,ODrTpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCsTpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC0TA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC2TpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwTtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC2TA,wB;QAMqD,OzPzPpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC4TpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCgUA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiUpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkUtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCqUA,wB;QAMqD,ODrUpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCsUpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC0UA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC2UpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwUtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC2UA,wB;QAMqD,OzPzPpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC4UpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCgVA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiVpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkVtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCqVA,wB;QAMqD,ODrVpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCsVpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC0VA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC2VpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwVtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC2VA,wB;QAMqD,OzPzPpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC4VpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCgWA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiWpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkWtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCqWA,wB;QAMqD,ODrWpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KCsWpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC0WA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KC2WpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfwWtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MC2WA,wB;QAMqD,OzPzPpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,KAAL,GAAiB,KAAtB,CD5KiB,EC4KjB,cAnLqC,KAmL3B,KAAL,GAAiB,KAAtB,CD5KiB,C;O;KC4WpD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCgXA,wB;QAMoD,OhFhFnDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ECAkB,KDAlB,C;O;KCiXpD,C;4FAOA,yB;MAIA,aAS6D,0B;MAT7D,+B;Me5KA,oD;MfAA,wB;QAMsD,OeAA,ef+KjB,eAAW,oBAAL,SAAK,CAAL,YAAN,Ce/KiB,EfAmB,KeAnB,C;O;KfkXtD,C;4FAOA,yB;MAIA,6B;MDzKA,kD;MCqXA,wB;QAMqD,ODrXpDE,WCqMjB,cAAU,SAAL,GAAiB,KAAtB,CDrMiB,ED2KjB,cEIqC,KFJ3B,

BAAN,OjCgUf,C;;QiChU8C,MAAM,2BAAuB,YAAM,WAA7B,C;K;;2CAG7F,mB;MAIS,Q;MAAL,IAAI,eAAC,0EAAD,SAAJ,C;QAAkC,OAAO,K;MAEzC,OAAe,WAAR,YAAQ,EAAS,OjCsNS,KiCtNlB,C;K;gDAGnB,oB;MACY,Q;MAA2B,gBAA3B,gE;MAA2B,c;;QjB0nDvB,U;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,aAAO,I;UAAP,e;;QACrB,6B;QAAhB,IAAgB,OAAgB,gBAAhB,hB,C;UAAgB,2B;UAAiB,1nD6B,2BjB0nDR,OiB1nDQ,S;UAAA,W;YAAwB,oBAAR,YAAQ,EjB0nDhC,OhBx6CA,KiClNgC,C;;UjB0nD/C,IAAI,OAAJ,C;YAAyB,aAAO,K;YAAP,e;;;QAAC,QAAC/C,aAAO,I;;;MiB3nDH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,kBAAqB,C;K;;IA/CvD,uC;MAAA,qD;MACgC,wBAAK,eAAL,IAAK,QAAL,CAAK,C;MAAL,IAAK,MADrC,C;MACgC,qCAAK,C

X,CAAhB,C;QmN+MnD,InN5N+D,amN4N3D,MnN5N6E,KAAlB,EmN4NlD,YnN5NgF,KAA9B,CmN4N3D,IA
AJ,C;UAA2B,OAAO,I;;MAGtC,OAAO,M;K;I3N9RX,6B;MACkD,OAAuB,0BAAtB,KAAO,WAAe,EAAU,KAA
O,WAAjB,C;K;IACzE,8B;MACqD,OAAC,gCAAuB,iBAAU,gCAAV,C;K;IAE7E,4B;MACoD,ORiZZ,SAvGI,oB
Q1SS,ER0Sd,KAAK,CAAL,iBQ1Sc,KR0ST,oBQ1SuB,ER0S5B,KAAK,CAAL,iBQ1Sc,CRiZH,QAAV,C;K;IQhZ
xC,+B;MACuD,OR+Yf,SAvGI,oBQxSY,ERwSjB,KAAK,CAAL,iBQxSiB,QRwSZ,oBQxS0B,ERwS/B,KAAK,C
AAL,iBQxSiB,CR+YN,QAAV,C;K;IQ1YxC,6B;MAEI,eAAe,EQkSoB,K;MRjSnC,cAAc,EQiSqB,K;MRhSnC,IA
AI,qBAAU,CAAd,C;QACI,OQ6C+D,aR7CpD,EQ6CsE,KAAlB,ER7C/C,EQ6C6E,KAA9B,CR7CpD,IAAJ,GAAAa,
aAAb,GAA2B,a;;MAItC,IAAI,uBAAY,CAAhB,C;QACI,OAAO,UAAM,aAAW,OAAX,CAAN,C;;MAIX,eAAiB,
4BAAc,CAAd,CAAD,KAAoB,OAApB,CAAD,WAAkC,CAAlC,C;MACf,UAAU,kBAAW,kBAAW,OAAX,CAAA
X,C;MACV,OAAO,UAAM,iCQkCsD,aAAkB,CRlCzD,UAAM,GAAN,CQkCyD,MAAlB,EAA8B,CRlCvD,UAA
M,OAAN,CQkCuD,MAA9B,CRlCvC,KAAJ,GAAkC,CAAlC,GAAyC,CAApD,EAAN,C;K;IAIX,gC;MAKe,Q;M
AHX,eAAe,EQ8QoB,K;MR7QnC,cAAc,EQ6CqB,K;MR5QnC,IAAI,qBAAU,CAAd,C;QACW,IQyBwD,aRzBpD,
EQyBsE,KAAlB,ERzB/C,EQyB6E,KAA9B,CRzBpD,IAAJ,C;UACH,S;;UAEA,OQgDgD,URhDhD,EQgD2D,KA
AK,URhD3D,EQgDuE,KAAZ,CAAhB,C;;QRnDpD,W;;MAQJ,IAAI,uBAAY,CAAhB,C;QACI,OAAO,UAAM,gB
AAW,OAAX,CAAN,C;;MAIX,eAAiB,4BAAc,CAAd,CAAD,KAAoB,OAApB,CAAD,WAAkC,CAAlC,C;MACf,
UAAU,kBAAW,kBAAW,OAAX,CAAX,C;MACV,OAAO,UAAM,aQUsD,aAAkB,CRV9D,UAAM,GAAN,CQU8D,MAAlB,
EAA8B,CRV5D,UAAM,OAAN,CQU4D,MAA9B,CRV5C,KAAJ,GAAkC,OAAlC,CAAN,CAAN,C;K;
IAGX,yB;MAEI,IAAE,QAAF,CAAE,CAAF,C;QADyC,OAC5B,W;;QACb,SRwSuC,aQxSlC,4BAAK,URwS0C,K
AAb,CQxSvC,C;UAFyC,OAEP,4BAAK,U;;UACvC,SRuSuC,aQvSlC,4BAAK,URuS0C,KAAb,CQvSvC,C;YAHy
C,OAGP,4BAAK,U;eACvC,SAAK,UAAL,C;YAJyC,ORkVN,SQ9UX,YAAF,CAAE,CR8UW,C;;YQlVM,ORgBY
,SAAU,CAkU5B,SQ7UP,YAAnB,IAAI,UAAe,CR6UO,CAlU4B,MAAK,GAAW,CAkU5C,SQ7UY,UR6UZ,CAlU
4C,MAAX,IAAf,C;;;;K;IQRzD,0B;MAEI,IAAE,QAAF,CAAE,CAAF,C;QAD2C,OAC9B,2B,SQkSuC,cRl
SlC,6BAAM,UQkS0C,KAAd,CRlSvC,C;UAF2C,OAER,6BAAM,U;;UACzC,SQiSuC,cRjSlC,6BAAM,UQiS0C,K
AAd,CRjSvC,C;YAH2C,OAGR,6BAAM,U;eACzC,4C;YAJ2C,OQwVL,URpVd,uBAAF,CAAE,CQoVc,C;;YRxV
K,OQUY,UAAW,CA8U5B,URjVF,uBAA3B,IAAI,oBAAuB,CQiVE,CA9U4B,MAAK,GAAW,CRHzB,gCQGyB,
MAAX,CAAhB,C;;;;K;IRC3D,yB;MAC4C,QAAC,CAAqB,GAAf,UAAP,IAAmC,CAAC,MAAO,EAAW,IAAJ,E
AAf,IAAgC,C;K;IAE/G,0B;MAC8C,OAAC,qBAAO,EAAP,CAAW,WAAZ,GAAyB,IAAzB,GAAgC,C,YAAjC,W;K
;IAG9C,0B;MAA8C,uBAAc,CAAd,EAAiB,EAAjB,C;K;IAE9C,kC;MACI,IAAI,gBAAK,CAAT,C;QAAY,OAAS,
WAAF,CAAE,CAAF,EAAS,IAAT,EAAe,qBAAO,CAAP,EAAU,yBAAa,IAAb,EAAmB,WAAnB,wB,CAAxB,C;
MACf,UAAU,WAAI,sCAAW,IAAX,EAAiB,EAAjB,C;MACV,OAAO,UAAM,iBAAO,IAAP,EAAa,C;QACtB,uCAAO,IAAP,
AY,CAAZ,E;;MAEJ,OAAgB,WAAT,QAAS,EAAT,IAAS,CAAT,GAAgB,8BAAhB,WAAgB,EAAhB,IAAgB,EAAhB,C;K;I4N1F
zC,qC;K;;;;

;;;

1.94 automation 1.11-8

1.94.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* dk.brics.automaton
*
* Copyright (c) 2001-2011 Anders Moeller
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
*   derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/BasicAutomata.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/TransitionComparator.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/DatatypesAutomatonProvider.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/Automaton.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/Datatypes.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/SpecialOperations.java

* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/BasicOperations.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/StatePair.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/MinimizationOperations.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/State.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/ShuffleOperations.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/RunAutomaton.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/RegExp.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/AutomatonProvider.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/Transition.java
No license file was found, but licenses were detected in source scan.

/*
* dk.brics.automaton - AutomatonMatcher
*
* Copyright (c) 2008-2011 John Gibson
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
*   derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-

jar/dk/brics/automaton/AutomatonMatcher.java

1.95 guava 31.0.1-jre

1.95.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for web specializations of methods of {@code Floats}. Intended to be empty for regular
* version.
*/
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/FloatsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2007 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/*
  * This following method is a modified version of one found in
```

```
 * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/test/tck/AbstractExecutorServiceTest.java?revision=1.30
 * which contained the following notice:
 *
 * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to
 * the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/
 *
 * Other contributors include Andrew Wright, Jeffrey Hayes, Pat Fisher, Mike Judd.
 */
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/MoreExecutors.java
No license file was found, but licenses were detected in source scan.

```
/*
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/publicdomain/zero/1.0/
*/
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/LongAdder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AtomicDoubleArray.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Striped64.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/Striped64.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/LongAdder.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
```

* Holder for web specializations of methods of {@code Doubles}. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/DoublesMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AtomicLongMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/GwtTransient.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2013 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MultimapBuilder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMapEntry.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/MoreFiles.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/SubscriberRegistry.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/MoreObjects.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/Subscriber.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ListenerCallQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/Dispatcher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/TrustedListenableFutureTask.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/Quantiles.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2007 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0

*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Functions.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Objects.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FinalizableSoftReference.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/AllowConcurrentEvents.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/DeadEvent.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ExecutionList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/MultiInputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/HashBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/LittleEndianDataInputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ListenableFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Charsets.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/EventBus.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Defaults.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Predicates.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Closeables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/Subscribe.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/io/CharStreams.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/AsyncEventBus.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Supplier.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/LineBuffer.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Preconditions.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Resources.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/LineReader.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/LittleEndianDataOutputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Throwables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Predicate.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/ByteStreams.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FinalizableReferenceQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Files.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FinalizableWeakReference.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/AbstractIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Primitives.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FinalizableReference.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/CountingOutputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FinalizablePhantomReference.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/EnumMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/CountingInputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Flushables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/DirectExecutor.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Interners.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Suppliers.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Function.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2012 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/ByteSource.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/AbstractByteHasher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/AbstractInvocationHandler.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/Parameter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ServiceManager.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/StandardSystemProperty.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/StatsAccumulator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CartesianList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/LongAddable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/TypeToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/html/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/ChecksumHashFunction.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/ByteSink.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/TypeCapture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/LinearTransformation.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/FilteredKeyMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/ClassPath.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/LongAddables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/xml/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/BaseEncoding.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/SmoothRateLimiter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/SipHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/MutableTypeToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/CharSource.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/Stats.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableRangeMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/PairedStatsAccumulator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/FileWriteMode.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableRangeSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/CharSink.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/LongAddable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/LongAddables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/RateLimiter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/PairedStats.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/Invokable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/ImmutableTypeToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Closer.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ListenableScheduledFuture.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2010 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/annotations/Beta.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ListeningExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ThreadFactoryBuilder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Ascii.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Equivalence.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Strings.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/UncaughtExceptionHandlers.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Atomics.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ContiguousSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedLists.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/package-

info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingBlockingQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Monitor.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/annotations/package-info.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2007 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Multiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Iterables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MapDifference.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Sets.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/HashMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TreeMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MutableClassToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SetMultimap.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Multisets.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/BiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingSortedMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ReverseOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractMapEntry.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ByFunctionOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Ordering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingMapEntry.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/EnumHashBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Interner.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingObject.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Multimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingListIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/LinkedListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ExplicitOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ComparatorOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractMapBasedMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingSortedSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Lists.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularImmutableSet.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/EnumBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/LinkedHashMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Iterators.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/NaturalOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Multimaps.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/NullsFirstOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/NullsLastOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/UsingToStringOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Maps.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ClassToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ConcurrentHashMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ArrayListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TreeMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingConcurrentMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractMapBasedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingCollection.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SingletonImmutableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractSortedSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/HashMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/LexicographicalOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CompoundOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Synchronized.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ReverseNaturalOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/LinkedHashMultiset.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2017 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/ImmutableIntArray.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingCondition.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingLock.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/ImmutableDoubleArray.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/AbstractHashFunction.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/ImmutableLongArray.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/LinkedHashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/DirectedNetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ElementOrder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/DirectedMultiNetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ImmutableValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CollectCollectors.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/MutableValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractGraphBuilder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/StandardMutableGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMultisetGwtSerializationDependencies.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ArrayListMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/GraphBuilder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/StandardMutableValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/Comparators.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/EndpointPairIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ValueGraphBuilder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/MapRetrievalCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ForwardingGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MoreCollectors.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/UndirectedMultiNetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/GraphConstants.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ForwardingValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/StandardMutableNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/StandardNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RangeGwtSerializationDependencies.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/StandardValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/MapIteratorCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractUndirectedNetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/NetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/HashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/MultiEdgesConnecting.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/UndirectedNetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/NetworkBuilder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/graph/ForwardingNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/EdgesConnecting.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractDirectedNetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/GraphConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/DirectedGraphConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/UndirectedGraphConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/EndpointPair.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2019 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Internal.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Platform.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2007 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
```

```
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
  * Returns an array containing all of the elements in the specified collection. This method
  * returns the elements in the order they are returned by the collection's iterator. The returned
  * array is "safe" in that no references to it are maintained by the collection. The caller is
  * thus free to modify the returned array.
  *
  * <p>This method assumes that the collection size doesn't change while the method is running.
  *
  * <p>TODO(kevinb): support concurrently modified collections?
  *
  * @param c the collection for which to return an array of elements
  */
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ObjectArrays.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2012 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/*
  * This method was rewritten in Java from an intermediate step of the Murmur hash function in
  * http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
  * following header:
  *
  * MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
  * hereby disclaims copyright to this source code.
  */
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/SmallCharMatcher.java

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2015 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you
* may not use this file except in compliance with the License.  You may
* obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
* implied.  See the License for the specific language governing
* permissions and limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Streams.java

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2012 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CompactHashSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/FilteredMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/DescendingImmutableSortedSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractNavigableMap.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/FilteredSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/FilteredKeySetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingImmutableList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingImmutableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CompactLinkedHashSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/UnmodifiableSortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/EvictingQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CompactHashMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TreeTraverser.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/DescendingMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableEnumMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/FilteredEntryMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularImmutableAsList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingNavigableMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TransformedListIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/FilteredKeyListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TreeRangeMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CompactLinkedHashMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingBlockingDeque.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/FilteredEntrySetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RangeMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingDeque.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractSortedKeySortedSetMultimap.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AllEqualOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingNavigableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingBlockingDeque.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TransformedIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedMultisetBridge.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingImmutableMap.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2013 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/TypeVisitor.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/SubscriberExceptionHandler.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/VerifyException.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Verify.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/thirdparty/publicsuffix/PublicSuffixType.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/FilteredMultimapValues.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/SubscriberExceptionContext.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/CharSequenceReader.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/WrappingScheduledExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/HashingInputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Utf8.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Runnables.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/GeneralRange.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularImmutableSortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableSortedMultisetFauxverideShim.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractRangeSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedIterable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Count.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableSortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedIterables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingSortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RangeSet.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2015 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/Platform.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/ReaderInputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/InterruptibleTask.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/AsyncCallable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/LittleEndianByteArray.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ConsumingQueueIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/MacHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/CombinedFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/AggregateFutureState.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/FarmHashFingerprint64.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either

express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/ImmutableSupplier.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/JdkBackedImmutableMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ExecutionSequencer.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/JdkBackedImmutableSet.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/AbstractStreamingHasher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/Cache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/AbstractLoadingCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/CacheBuilderSpec.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/BloomFilter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/UnsignedLong.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/ForwardingLoadingCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/DescendingImmutableSortedMultiset.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/HostAndPort.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/PrimitiveSink.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/IntMath.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Hasher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/LoadingCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/ParseRequest.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Hashing.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/DoubleUtils.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/Weigher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/UnsignedInteger.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractScheduledService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractListeningExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/UnsignedInts.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ExecutionError.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Funnel.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/TypeParameter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/BloomFilterStrategies.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/MediaType.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Uninterruptibles.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/HttpHeaders.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/RemovalCause.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/package-info.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/MathPreconditions.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Ticker.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Absent.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/EmptyContiguousSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/DoubleMath.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/RemovalListener.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractSortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Enums.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/AbstractCompositeHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Present.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Funnels.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/UnsignedLongs.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/HashingOutputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Optional.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/BigIntegerMath.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/AbstractHasher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/BoundType.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TreeRangeSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/RemovalListeners.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/ForwardingCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/FutureCallback.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/HashCode.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/package-info.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Murmur3_128HashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/LongMath.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/AbstractNonStreamingHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/MessageDigestHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Murmur3_32HashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Queues.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/AbstractCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingListeningExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularImmutableMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Crc32cHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/WrappingExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/UncheckedExecutionException.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/HashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/CacheStats.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/RemovalNotification.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AsyncFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularContiguousSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ListeningScheduledExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FunctionalEquivalence.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/Types.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/CacheLoader.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/package-info.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TopKSelector.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/Graphs.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/PredecessorsFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ImmutableNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/Graph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/MutableGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/Network.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/graph/MutableNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/SuccessorsFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/InsecureRecursiveDeleteException.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ImmutableGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/RecursiveDeleteOption.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
  * Not supported. <b>You are attempting to create a map that may contain a non-{@code Comparable}
  * key.</b> Proper calls will resolve to the version in {@code ImmutableSortedMap}, not this dummy
  * version.
  *
  * @throws UnsupportedOperationException always
  * @deprecated <b>Pass a key of type {@code Comparable} to use {@link
  *     ImmutableSortedMap#of(Comparable, Object)}.</b>
  */
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableSortedMapFauxverideShim.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2016 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
```

*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/JdkPattern.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/CommonPattern.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/CommonMatcher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/PatternCompiler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for web specializations of methods of {@code Ints}. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/IntsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2006 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/FuturesGetChecked.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/UncheckedTimeoutException.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/TimeLimiter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/FluentFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ImmediateFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/CharEscaper.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/TypeToken.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/PatternFilenameFilter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/annotations/VisibleForTesting.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/SimpleTimeLimiter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/FakeTimeLimiter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/AppendableWriter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Futures.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractCatchingFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/TimeoutFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/GwtFuturesCatchingSpecialization.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/CharEscaperBuilder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractTransformFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/CollectionFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/CaseFormat.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AggregateFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/GwtFluentFutureCatchingSpecialization.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableSortedAsList.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2005 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/Reflection.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingListenableFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Service.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/CacheBuilder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MapMakerInternalMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/annotations/GwtCompatible.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/annotations/GwtIncompatible.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/LocalCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractExecutionThreadService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractIdleService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/ReferenceEntry.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/InternetDomainName.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/html/HtmlEscapers.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Cut.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Platform.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingFluentFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/Escapers.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/SignedBytes.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/UnsignedBytes.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/HostSpecifier.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/ArrayBasedCharEscaper.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Splitter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/Platform.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MapMaker.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/DenseImmutableTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/UrlEscapers.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/ByteArrayDataOutput.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularImmutableTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Callables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/ArrayBasedEscaperMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/TypeResolver.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/ByteArrayDataInput.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/ArrayBasedUnicodeEscaper.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SparseImmutableTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/xml/XmlEscapers.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/LineProcessor.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/JdkFutureAdapters.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/SettableFuture.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/ByteProcessor.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2015 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableBiMapFauxverideShim.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CollectSpliterators.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/JdkBackedImmutableBiMap.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/IndexedImmutableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/BaseImmutableMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/JdkBackedImmutableMap.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2010 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MinMaxPriorityQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingImmutableCollection.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractSequentialIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/UnmodifiableListIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingSortedSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RowSortedTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedMapDifference.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableClassToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/DiscreteDomain.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularImmutableSortedSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableSortedSetFauxverideShim.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ForwardingTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableEnumSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableAsList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractIndexedListIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/EmptyImmutableSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SingletonImmutableTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TableCollectors.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularImmutableList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ComparisonChain.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableSortedMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ArrayTable.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ComputationException.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SingletonImmutableList.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2021 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/xml/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/primitives/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/xml/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/html/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/html/ParametricNullness.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express

* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for web specializations of methods of {@code Shorts}. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/ShortsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/*
 * This method was written by Doug Lea with assistance from members of JCP JSR-166 Expert Group
 * and released to the public domain, as explained at
 * http://creativecommons.org/licenses/publicdomain
 *
 * As of 2010/06/11, this method is identical to the (package private) hash method in OpenJDK 7's
 * java.util.HashMap class.
 */

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Striped.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2017 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*

* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractBaseGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ClosingFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/BaseGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/Traverser.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for extra methods of {@code Objects} only in web. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/ExtraObjectsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/BigDecimalMath.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/OverflowAvoidingLockSupport.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Java8Compatibility.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/ToDoubleRounder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Java8Compatibility.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Java8Compatibility.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Outer class that exists solely to let us write {@code Partially.GwtIncompatible} instead of plain
* {@code GwtIncompatible}. This is more accurate for {@link Futures#catching}, which is available
* under GWT but with a slightly different signature.
*
* <p>We can't use {@code PartiallyGwtIncompatible} because then the GWT compiler wouldn't recognize
* it as a {@code GwtIncompatible} annotation. And for {@code Futures.catching}, we need the GWT
* compiler to autostrip the normal server method in order to expose the special, inherited GWT

* version.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Partially.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ServiceManagerBridge.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2021 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/NullnessCasts.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/NullnessCasts.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/NullnessCasts.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2008 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Platform.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMapEntrySet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMapValues.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Tables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Collections2.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/HashBasedTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CollectPreconditions.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/thirdparty/publicsuffix/PublicSuffixPatterns.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMapKeySet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/StandardRowSortedTable.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularImmutableMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/PeekingIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Table.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableEntry.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularImmutableBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SingletonImmutableBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Serialization.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableSortedSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/StandardTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/EmptyImmutableListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableCollection.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Range.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TreeBasedTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/UnmodifiableIterator.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2008 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/internal/Finalizer.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Booleans.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/thirdparty/publicsuffix/TrieParser.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Ints.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ListenableFutureTask.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Chars.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Shorts.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/Escaper.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Floats.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Stopwatch.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/FileBackedOutputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/CharMatcher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/UnicodeEscaper.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/FluentIterable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/MultiReader.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Converter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Doubles.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/InetAddresses.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Longs.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Bytes.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Joiner.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/SequentialExecutor.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/PercentEscaper.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedMultisets.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/IncidentEdgeSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CompactHashing.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2008 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/*
 * This method was rewritten in Java from an intermediate step of the Murmur hash function in
 * http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
 * following header:
 *
 * MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
 * hereby disclaims copyright to this source code.
 */
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Hashing.java

1.96 junit-5-bill-of-materials 5.8.2

1.96.1 Available under license :

```
import java.io.File
import java.net.URI

data class License(val name: String, val url: URI, val headerFile: File)
```

Apache License
==========

Version 2.0, January 2004
<<https://www.apache.org/licenses/>>

Terms and Conditions for use, reproduction, and distribution

1. Definitions

License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently

incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby
grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free,
irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such
Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby
grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free,
irrevocable (except as stated in this section) patent license to make, have
made, use, offer to sell, sell, import, and otherwise transfer the Work, where
such license applies only to those patent claims licensable by such Contributor
that are necessarily infringed by their Contribution(s) alone or by combination
of their Contribution(s) with the Work to which such Contribution(s) was
submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a
Contribution incorporated within the Work constitutes direct or contributory
patent infringement, then any patent licenses granted to You under this License
for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof
in any medium, with or without modifications, and in Source or Object form,
provided that You meet the following conditions:

* **(a)** You must give any other recipients of the Work or Derivative Works a copy of
this License; and
* **(b)** You must cause any modified files to carry prominent notices stating that You
changed the files; and
* **(c)** You must retain, in the Source form of any Derivative Works that You distribute,
all copyright, patent, trademark, and attribution notices from the Source form
of the Work, excluding those notices that do not pertain to any part of the
Derivative Works; and
* **(d)** If the Work includes a NOTICE text file as part of its distribution, then any
Derivative Works that You distribute must include a readable copy of the
attribution notices contained within such NOTICE file, excluding those notices
that do not pertain to any part of the Derivative Works, in at least one of the
following places: within a NOTICE text file distributed as part of the
Derivative Works; within the Source form or documentation, if provided along
with the Derivative Works; or, within a display generated by the Derivative
Works, if and wherever such third-party notices normally appear. The contents of
the NOTICE file are for informational purposes only and do not modify the

License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has

been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to
offer, and charge a fee for, acceptance of support, warranty, indemnity, or
other liability obligations and/or rights consistent with this License. However,
in accepting such obligations, You may act only on Your own behalf and on Your
sole responsibility, not on behalf of any other Contributor, and only if You
agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your
accepting any such warranty or additional liability.
Eclipse Public License - v 2.0
==========

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:
* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **b)** in the case of each subsequent Contributor:
 * **i)** changes to the Program, and
 * **ii)** additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations,

interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

3.1 If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
 * **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 * **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 * **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
 * **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
* **b)** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
Apache License
==========

Version 2.0, January 2004
<<https://www.apache.org/licenses/>>

Terms and Conditions for use, reproduction, and distribution

1. Definitions

License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor

that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

* **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
* **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
* **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
* **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of

any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate
notice, with the fields enclosed by brackets `[]` replaced with your own
identifying information. (Don't include the brackets!) The text should be
enclosed in the appropriate comment syntax for the file format. We also
recommend that a file or class name and description of purpose be included on
the same printed page as the copyright notice for easier identification within
third-party archives.

  Copyright [yyyy] [name of copyright owner]

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
Open Source Licenses
====================

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for
these subcomponents is subject to the terms and conditions of the
subcomponent's license, as noted in the LICENSE-<subcomponent>.md
files.
[[contributors]]
== Contributors

Browse the {junit5-repo}/graphs/contributors[current list of contributors] directly on GitHub.

1.97 fabric8-::-kubernetes-model-::-api-extensions 5.12.1

1.97.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.

You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815613_1654860972.37505/0/kubernetes-model-apiextensions-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-apiextensions/pom.xml
No license file was found, but licenses were detected in source scan.

```
/**
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*        http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1340815613_1654860972.37505/0/kubernetes-model-apiextensions-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/apiextensions/v1/JSONSchemaPropsOrArraySerDe.java
* /opt/cola/permits/1340815613_1654860972.37505/0/kubernetes-model-apiextensions-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/apiextensions/v1beta1/JSONSchemaPropsOrBoolSerDe.java
* /opt/cola/permits/1340815613_1654860972.37505/0/kubernetes-model-apiextensions-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/apiextensions/v1/JSONSchemaPropsOrBoolSerDe.java
* /opt/cola/permits/1340815613_1654860972.37505/0/kubernetes-model-apiextensions-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/apiextensions/v1beta1/JSONSchemaPropsOrStringArraySerDe.java
* /opt/cola/permits/1340815613_1654860972.37505/0/kubernetes-model-apiextensions-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/apiextensions/v1beta1/JSONSchemaPropsOrArraySerDe.java
* /opt/cola/permits/1340815613_1654860972.37505/0/kubernetes-model-apiextensions-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/apiextensions/v1/JSONSchemaPropsOrStringArraySerDe.java
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340815613_1654860972.37505/0/kubernetes-model-apiextensions-5-12-1-sources-jar/manifest.vm

1.98 spring-cloud-sleuth-autoconfigure 3.1.3

1.98.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2013-2018 the original author or authors.
~
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~
~   https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.
-->

Found in path(s):
* /opt/cola/permits/1519137129_1671710604.1743295/0/spring-cloud-sleuth-autoconfigure-3-1-3-jar/META-INF/maven/org.springframework.cloud/spring-cloud-sleuth-autoconfigure/pom.xml

1.99 asm-based-accessors-helper-used-by-json-smart 2.4.8

1.99.1 Available under license :

No license file was found, but licenses were detected in source scan.

<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>

Found in path(s):
* /opt/cola/permits/1340076313_1654762627.85568/0/accessors-smart-2-4-8-jar/META-INF/maven/net.minidev/accessors-smart/pom.xml
No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bnd-LastModified: 1644737816481
Build-Jdk: 11.0.14
Built-By: uriel
Bundle-Description: Java reflect give poor performance on getter sette
r an constructor calls, accessors-smart use ASM to speed up those cal
ls.
Bundle-DocURL: https://urielch.github.io/
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.txt
Bundle-ManifestVersion: 2
Bundle-Name: accessors-smart
Bundle-SymbolicName: net.minidev.accessors-smart
Bundle-Vendor: Chemouni Uriel
Bundle-Version: 2.4.8
Created-By: Apache Maven Bundle Plugin
Export-Package: net.minidev.asm;uses:="org.objectweb.asm";version="2.4
.8",net.minidev.asm.ex;version="2.4.8"
Import-Package: org.objectweb.asm;version="[8.0,10)",net.minidev.asm.e
x
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.8))"
Tool: Bnd-5.1.1.202006162103

Found in path(s):
* /opt/cola/permits/1340076313_1654762627.85568/0/accessors-smart-2-4-8-jar/META-INF/MANIFEST.MF

1.100 apache-log4j 2.17.2

1.100.1 Available under license :

Apache Log4j Docker Library
Copyright 1999-2022 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.101 brave-instrumentation-kafka-clients 5.13.9

1.101.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0
* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed
by Square, Inc in the Moshi Library:

* Copyright (C) 2010 Google Inc.
* License: Apache License v2.0
* Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

 * License: Apache License v2.0
 * Homepage: https://github.com/raphw/weak-lock-free

1.102 project-lombok 1.18.24

1.102.1 Available under license :

Copyright (C) 2009-2021 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

==========

Licenses for included components:

org.ow2.asm:asm
org.ow2.asm:asm-analysis
org.ow2.asm:asm-commons
org.ow2.asm:asm-tree
org.ow2.asm:asm-util
ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright

notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---

rzwitserloot/com.zwitserloot.cmdreader

Copyright 2010 Reinier Zwitserloot.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

projectlombok/lombok.patcher

Copyright (C) 2009-2021 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy

of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.103 junit-jupiter-junit-jupiter-params 5.8.2

1.103.1 Available under license :

Apache License
==========

Version 2.0, January 2004
<<https://www.apache.org/licenses/>>

Terms and Conditions for use, reproduction, and distribution

1. Definitions

License shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright
owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities
that control, are controlled by, or are under common control with that entity.
For the purposes of this definition, control means **(i)** the power, direct or
indirect, to cause the direction or management of such entity, whether by
contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the
outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising
permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby

grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

* **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
* **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
* **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
* **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Eclipse Public License - v 2.0
==========

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:
* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **b)** in the case of each subsequent Contributor:
 * **i)** changes to the Program, and
 * **ii)** additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

3.1 If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
 * **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 * **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 * **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
 * **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
* **b)** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
Open Source Licenses
==========

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.104 jsonassert 1.5.1

1.104.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/comparator/DefaultComparator.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/FieldComparisonFailure.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/Customization.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/RegularExpressionValueMatcher.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/json/JSONString.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/ValueMatcherException.java

* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/comparator/ArraySizeComparator.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/JSONAssert.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/ValueMatcher.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/comparator/AbstractComparator.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/JSONCompare.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/comparator/JSONCompareUtil.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/comparator/CustomComparator.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/JSONParser.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/comparator/JSONComparator.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/JSONCompareResult.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/ArrayValueMatcher.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/LocationAwareValueMatcher.java
* /opt/cola/permits/1519136915_1671710635.2038014/0/jsonassert-1-5-1-sources-1-jar/org/skyscreamer/jsonassert/JSONCompareMode.java

1.105 testcontainers-database-commons 1.17.1

1.105.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2014 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and

* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136972_1671725245.9734879/0/database-commons-1-17-1-sources-jar/org/testcontainers/ext/ScriptUtils.java

1.106 reactive-relational-database-connectivity-spi 0.9.1.RELEASE

1.106.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Reactive Relational Database Connectivity

Copyright 2017-2021 the original author or authors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.107 jooq 3.16.4

1.107.1 Available under license :

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Other licenses:

Commercial licenses for this work are available. These replace the above
ASL 2.0 and offer limited warranties, support, maintenance, and commercial
database integrations.

For more information, please visit: http://www.jooq.org/licenses

1.108 opentelemetry-java 1.14.0-alpha

1.108.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.109 brave-instrumentation-messaging-adapters 5.13.9

1.109.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

 * Copyright (C) 2008 The Guava Authors
 * License: Apache License v2.0
 * Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed
by Square, Inc in the Moshi Library:

 * Copyright (C) 2010 Google Inc.
 * License: Apache License v2.0
 * Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

 * License: Apache License v2.0
 * Homepage: https://github.com/raphw/weak-lock-free

1.110 brave-instrumentation-apache-httpclient-v 5.13.9

1.110.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0

* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed
by Square, Inc in the Moshi Library:

* Copyright (C) 2010 Google Inc.
* License: Apache License v2.0
* Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

* License: Apache License v2.0
* Homepage: https://github.com/raphw/weak-lock-free

1.111 spring-security-rsa 1.0.10.RELEASE

1.111.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013-2014 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1337070288_1654119493.5792491/0/spring-security-rsa-1-0-10-release-sources-1-jar/org/springframework/security/rsa/crypto/RsaKeyHelper.java
* /opt/cola/permits/1337070288_1654119493.5792491/0/spring-security-rsa-1-0-10-release-sources-1-jar/org/springframework/security/rsa/crypto/RsaRawEncryptor.java
* /opt/cola/permits/1337070288_1654119493.5792491/0/spring-security-rsa-1-0-10-release-sources-1-jar/org/springframework/security/rsa/crypto/RsaKeyHolder.java
* /opt/cola/permits/1337070288_1654119493.5792491/0/spring-security-rsa-1-0-10-release-sources-1-jar/org/springframework/security/rsa/crypto/RsaSecretEncryptor.java
* /opt/cola/permits/1337070288_1654119493.5792491/0/spring-security-rsa-1-0-10-release-sources-1-jar/org/springframework/security/rsa/crypto/KeyStoreKeyFactory.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2011 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1337070288_1654119493.5792491/0/spring-security-rsa-1-0-10-release-sources-1-jar/org/springframework/security/rsa/crypto/CipherUtils.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1337070288_1654119493.5792491/0/spring-security-rsa-1-0-10-release-sources-1-jar/org/springframework/security/rsa/crypto/RsaAlgorithm.java

1.112 fabric8-::-kubernetes-model-::-events 5.12.1

1.112.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*         http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
```

Found in path(s):
* /opt/cola/permits/1340816446_1654860848.8812537/0/kubernetes-model-events-5-12-1-sources-jar/manifest.vm
No license file was found, but licenses were detected in source scan.

```
<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

        http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->
```

Found in path(s):
* /opt/cola/permits/1340816446_1654860848.8812537/0/kubernetes-model-events-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-events/pom.xml

1.113 spring-cloud-starter 3.1.3

1.113.1 Available under license :

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1519136866_1671710616.7249413/0/spring-cloud-starter-3-1-3-2-jar/META-INF/maven/org.springframework.cloud/spring-cloud-starter/pom.xml

1.114 reactor-test-support 3.4.24

1.114.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2018-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/MessageFormatter.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/util/TestLogger.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2021 VMware Inc. or its affiliates, All Rights Reserved.
*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/subscriber/DefaultConditionalTestSubscriber.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/subscriber/TestSubscriber.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/subscriber/TestSubscriberBuilder.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/subscriber/ConditionalTestSubscriber.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/subscriber/package-info.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/subscriber/DefaultTestSubscriber.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2017-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/publisher/TestPublisher.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-

jar/reactor/test/publisher/DefaultTestPublisher.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/publisher/PublisherProbe.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/StepVerifierOptions.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/publisher/ColdTestPublisher.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2011-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/StepVerifierExtensions.kt
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/scheduler/package-info.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/publisher/package-info.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/package-info.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2019-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/ValueFormatters.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2017-2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/StepVerifier.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/scheduler/VirtualTimeScheduler.java
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/DefaultStepVerifierBuilder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2016-2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/util/RaceTestUtils.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2020-2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137169_1671721463.2732697/0/reactor-test-3-4-24-sources-jar/reactor/test/util/LoggerUtils.java

1.115 lz4-and-xxhash 1.8.0

1.115.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 Rei Odaira and the lz4-java contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4DecompressorWithLength.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4CompressorWithLength.java
No license file was found, but licenses were detected in source scan.

<!--
  Copyright 2020 Adrien Grand and the lz4-java contributors.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->

Found in path(s):
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/overview.html
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/package.html
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/package.html
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/util/package.html
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 Linnaea Von Lavia and the lz4-java contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/XXHash64JNI.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/XXHash64.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/StreamingXXHash64JNI.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/StreamingXXHash64.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/AbstractStreamingXXHash64Java.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 The Apache Software Foundation and the lz4-java contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4FrameInputStream.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4FrameOutputStream.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 Adrien Grand and the lz4-java contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/XXHash32.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4BlockOutputStream.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/util/UnsafeUtils.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4BlockInputStream.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/util/ByteBufferUtils.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4Decompressor.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/util/Utils.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4JNICompressor.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4Compressor.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/XXHashJNI.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/util/SafeUtils.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/StreamingXXHash32.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/StreamingXXHash32JNI.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/XXHash32JNI.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4SafeDecompressor.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/util/Native.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4Exception.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4Factory.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4UnsafeUtils.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4JNI.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/AbstractStreamingXXHash32Java.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/XXHashFactory.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4ByteBufferUtils.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-

jar/net/jpountz/lz4/LZ4FastDecompressor.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4JNISafeDecompressor.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4UnknownSizeDecompressor.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4Constants.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/xxhash/XXHashConstants.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4JNIFastDecompressor.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4Utils.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4HCJNICompressor.java
* /opt/cola/permits/1340816063_1654804394.0569026/0/lz4-java-1-8-0-sources-jar/net/jpountz/lz4/LZ4SafeUtils.java

1.116 jetbrains-kotlin-kotlin-stdlib-jdk7 1.6.21

1.116.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2017 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1343419565_1655661636.7344434/0/kotlin-stdlib-jdk7-1-6-21-sources-jar/kotlin/AutoCloseable.kt

1.117 apache-log4j-slf4j-binding 2.17.2

1.117.1 Available under license :

Apache Log4j Core
Copyright 1999-2012 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate

as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify

the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include

the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache Log4j
Copyright 1999-2021 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

TypeUtil.java
Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

picocli (http://picocli.info)
Copyright 2017 Remko Popma
Apache Log4j
Copyright 1999-2012 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

Maven Wrapper Jar
Copyright 2016-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.118 jacoco 0.8.7

1.118.1 Available under license :

Found license 'Eclipse Public License 1.0' in 'This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at'
Found license 'Eclipse Public License 1.0' in 'under the terms and conditions of the Eclipse Public License Version 2.0'

1.119 functionality-for-the-reactor-netty-library 1.0.21

1.119.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2011-2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-

jar/reactor/netty/http/websocket/WebsocketOutbound.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientRequest.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpPredicate.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerRequest.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerResponse.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/package-info.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/package-info.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/package-info.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientResponse.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/websocket/WebsocketInbound.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/Cookies.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/DefaultHttpServerRoutes.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerRoutes.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpInfos.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2019-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerMetricsRecorder.java

* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerMetricsHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/MicrometerHttpMetricsRecorder.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/MicrometerHttpClientMetricsRecorder.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientMetricsHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/MicrometerHttpServerMetricsRecorder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2018-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/Http2StreamBridgeServerHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2021-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/ContextAwareHttpServerMetricsHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/MicrometerHttp2ConnectionProviderMeterRegistrar.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/AbstractHttpClientMetricsHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/ContextAwareHttpClientMetricsHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/AbstractHttpServerMetricsHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/Http2Pool.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2020-2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/BaseAccessLogHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/FailedHttpClientRequest.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/Http2SettingsSpec.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLog.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogArgProvider.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogFactory.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/DefaultHttpForwardedHeaderHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-

jar/reactor/netty/http/server/logging/package-info.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/websocket/package-info.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/WebsocketClientSpec.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/websocket/WebsocketSpec.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/WebsocketServerSpec.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/WebsocketServerSpecImpl.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogArgProviderH2.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/WebsocketClientSpecImpl.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpTrafficHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AbstractAccessLogArgProvider.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogHandlerFactory.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/websocket/WebsocketSpecImpl.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogArgProviderH1.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (c) 2011-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/WebsocketServerOperations.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientOperations.java

* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpOperations.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/RedirectClientException.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/WebsocketClientOperations.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpResources.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServer.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClient.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientFormEncoder.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerOperations.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpTrafficHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2020-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/Http2ConnectionProvider.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerConfig.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpConnectionProvider.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/Http2StreamBridgeClientHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientConfig.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2017-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientConnect.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2018-2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientState.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogHandlerH2.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpProtocol.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpRequestDecoderSpec.java

* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerState.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientTcpConfig.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogHandlerH1.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/SimpleCompressionHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerTcpConfig.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/ConnectionInfo.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/WebsocketFinalizer.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientSecure.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/Http2AllocationStrategy.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2017-2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerBind.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientForm.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/UriEndpointFactory.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/UriEndpoint.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientFinalizer.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2019-2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientInfos.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/ProxyProtocolSupportType.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpDecoderSpec.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpResponseDecoderSpec.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpMetricsRecorder.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-

jar/reactor/netty/http/server/HAProxyMessageDetector.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HAProxyMessageReader.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientMetricsRecorder.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/PrematureCloseException.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientSecurityUtils.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/Http2SslContextSpec.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/Http11SslContextSpec.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerFormDecoderProvider.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/ContextAwareHttpClientMetricsRecorder.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/ContextAwareHttpServerMetricsRecorder.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpRouteHandlerMetadata.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/ServerCookies.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/NonSslRedirectDetector.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/NonSslRedirectHandler.java
* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/ContextAwareHttpMetricsRecorder.java

* /opt/cola/permits/1519137122_1671725121.9089081/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerInfos.java

1.120 fabric8-::-kubernetes-model-::-storage-class 5.12.1

1.120.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*        http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
```

Found in path(s):
* /opt/cola/permits/1340816172_1654860375.1628633/0/kubernetes-model-storageclass-5-12-1-sources-jar/manifest.vm
No license file was found, but licenses were detected in source scan.

```
<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

        http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
```

-->

Found in path(s):
* /opt/cola/permits/1340816172_1654860375.1628633/0/kubernetes-model-storageclass-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-storageclass/pom.xml

1.121 mockito 4.5.1

1.121.1 Available under license :

OpenEJB :: Utils :: Mockito
Copyright 1999-2012 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but

not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their

Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with

the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.122 byte-buddy 1.12.12

1.122.1 Available under license :

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all

other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Copyright ${project.inceptionYear} - ${current.year} ${copyright.holder}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.123 kotlin-stdlib-common 1.6.21

1.123.1 Available under license :

Apache-2.0

1.124 jakarta-annotations-api 1.3.5

1.124.1 Available under license :

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE
PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION
OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License,

Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was

received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and

to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is

to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and

can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the

executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other

circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.

EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

> One line to give the program's name and a brief idea of what it does.
> Copyright (C) <year> <name of author>
>
> This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
>
> This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
>
> You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
    `show w'. This is free software, and you are welcome to redistribute
    it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

    Yoyodyne, Inc., hereby disclaims all copyright interest in the
    program `Gnomovision' (which makes passes at compilers) written by
    James Hacker.

    signature of Ty Coon, 1 April 1989
    Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module. An independent module is a module which is not derived from or

based on this library. If you modify this library, you may extend this
exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement
from your version.
[subs="normal"]
....
Specification: {doctitle}

Version: {revnumber}

ifeval::["{revremark}" != ""]
Status: {revremark}
endif::[]
ifeval::["{revremark}" == ""]
Status: Final Release
endif::[]

Release: {revdate}
....
Copyright (c) 2019 Eclipse Foundation.

=== Eclipse Foundation Specification License

By using and/or copying this document, or the Eclipse Foundation
document from which this statement is linked, you (the licensee) agree
that you have read, understood, and will comply with the following
terms and conditions:

Permission to copy, and distribute the contents of this document, or
the Eclipse Foundation document from which this statement is linked, in
any medium for any purpose and without fee or royalty is hereby
granted, provided that you include the following on ALL copies of the
document, or portions thereof, that you use:

* link or URL to the original Eclipse Foundation document.
* All existing copyright notices, or if one does not exist, a notice
 (hypertext is preferred, but a textual representation is permitted)
 of the form: "Copyright (c) [$date-of-document]
 Eclipse Foundation, Inc. <<url to this license>>"

Inclusion of the full text of this NOTICE must be provided. We
request that authorship attribution be provided in any software,
documents, or other items or products that you create pursuant to the
implementation of the contents of this document, or any portion
thereof.

No right to create modifications or derivatives of Eclipse Foundation
documents is granted pursuant to this license, except anyone may

prepare and distribute derivative works and portions of this document
in software that implements the specification, in supporting materials
accompanying such software, and in documentation of such software,
PROVIDED that all such works include the notice below. HOWEVER, the
publication of derivative works of this document for use as a technical
specification is expressly prohibited.

The notice is:

"Copyright (c) 2018 Eclipse Foundation. This software or
document includes material copied from or derived from [title and URI
of the Eclipse Foundation specification document]."

==== Disclaimers

THIS DOCUMENT IS PROVIDED "AS IS," AND THE COPYRIGHT
HOLDERS AND THE ECLIPSE FOUNDATION MAKE NO REPRESENTATIONS OR
WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE,
NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE
SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS
WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR
OTHER RIGHTS.

THE COPYRIGHT HOLDERS AND THE ECLIPSE FOUNDATION WILL NOT BE LIABLE
FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT
OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE
CONTENTS THEREOF.

The name and trademarks of the copyright holders or the Eclipse
Foundation may NOT be used in advertising or publicity pertaining to
this document or its contents without specific, written prior
permission. Title to copyright in this document will at all times
remain with copyright holders.
Notices for Jakarta Annotations

This content is produced and maintained by the Jakarta Annotations project.

* Project home: https://projects.eclipse.org/projects/ee4j.ca

Trademarks

Jakarta Annotations is a trademark of the Eclipse Foundation.

Declared Project Licenses

This program and the accompanying materials are made available under the terms
of the Eclipse Public License v. 2.0 which is available at

http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/common-annotations-api

Third-party Content

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.125 guava-listenablefuture-only 9999.0-empty-to-avoid-conflict-with-guava

1.125.1 Available under license :

Found license 'GNU Lesser General Public License' in '// This library is free software; you can redistribute it and/or // modify it under the terms of the GNU Lesser General Public // License as published by the Free Software Foundation; either // version 2.1 of the License, or (at your option) any later version. // This library is distributed in the hope that it will be useful, // but WITHOUT ANY WARRANTY; without even the implied warranty of // MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU // Lesser General Public License for more details. // You should have received a copy of the GNU Lesser General Public * This grammar is in the PUBLIC DOMAIN'

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence

the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run

that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any

warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to

exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the
source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not
compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the
Library, but is designed to work with the Library by being compiled or
linked with it, is called a "work that uses the Library". Such a
work, in isolation, is not a derivative work of the Library, and
therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library
creates an executable that is a derivative of the Library (because it
contains portions of the Library), rather than a "work that uses the
library". The executable is therefore covered by this License.
Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the

Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining

where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is

implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR

PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

 Yoyodyne, Inc., hereby disclaims all copyright interest in the
 library `Frob' (a library for tweaking knobs) written by James Random Hacker.

 <signature of Ty Coon>, 1 April 1990
 Ty Coon, President of Vice

That's all there is to it!
//
// checkstyle: Checks Java source code for adherence to a set of rules.
// Copyright (C) 2001-2020 the original author or authors.
//
// This library is free software; you can redistribute it and/or
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
//
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
// Lesser General Public License for more details.
//
// You should have received a copy of the GNU Lesser General Public
// License along with this library; if not, write to the Free Software
// Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
//

1.126 docker-java 3.2.13

1.126.1 Available under license :

Apache-2.0

1.127 jetbrains-kotlin-kotlin-stdlib-jdk8 1.6.21

1.127.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2017 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1343419559_1655661587.9846208/0/kotlin-stdlib-jdk8-1-6-21-sources-jar/kotlin/internal/jdk8/JDK8PlatformImplementations.kt
* /opt/cola/permits/1343419559_1655661587.9846208/0/kotlin-stdlib-jdk8-1-6-21-sources-jar/kotlin/streams/Streams.kt
* /opt/cola/permits/1343419559_1655661587.9846208/0/kotlin-stdlib-jdk8-1-6-21-sources-jar/kotlin/text/RegexExtensions.kt
* /opt/cola/permits/1343419559_1655661587.9846208/0/kotlin-stdlib-jdk8-1-6-21-sources-jar/kotlin/collections/Collections.kt

1.128 brave 5.13.9

1.128.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity

exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0
* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed

by Square, Inc in the Moshi Library:

 * Copyright (C) 2010 Google Inc.
 * License: Apache License v2.0
 * Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

 * License: Apache License v2.0
 * Homepage: https://github.com/raphw/weak-lock-free

1.129 java-vault-driver 5.1.0

1.129.1 Available under license :

No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2015 EclipseSource.
*
* Permission is hereby granted, free of charge, to any person obtaining a copy
* of this software and associated documentation files (the "Software"), to deal
* in the Software without restriction, including without limitation the rights
* to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
* copies of the Software, and to permit persons to whom the Software is
* furnished to do so, subject to the following conditions:
*
* The above copyright notice and this permission notice shall be included in all
* copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
***/

Found in path(s):
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/WritingBuffer.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/WritingBuffer_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/PrettyPrint.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-

master/src/test/java/com/bettercloud/vault/json/PrettyPrint_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/Json.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/Json_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/WriterConfig.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2013, 2015 EclipseSource.
*
* Permission is hereby granted, free of charge, to any person obtaining a copy
* of this software and associated documentation files (the "Software"), to deal
* in the Software without restriction, including without limitation the rights
* to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
* copies of the Software, and to permit persons to whom the Software is
* furnished to do so, subject to the following conditions:
*
* The above copyright notice and this permission notice shall be included in all
* copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
***/

Found in path(s):
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/ParseException_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/test/mocking/Mocking_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/JsonObject.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/TestUtil.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/JsonWriter_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/JsonString_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/JsonValue.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/JsonParser.java

* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/JsonWriter.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/ParseException.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/JsonObject_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/JsonLiteral.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/JsonNumber.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/JsonArray.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/JsonNumber_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/JsonParser_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/JsonLiteral_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/JsonValue_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/test/java/com/bettercloud/vault/json/JsonArray_Test.java
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/JsonString.java
No license file was found, but licenses were detected in source scan.

Vault Java Driver
==========

A zero-dependency Java client for the [Vault](https://www.vaultproject.io/) secrets management solution from HashiCorp.

This driver strives to implement Vault's full HTTP API, along with supporting functionality such as automatic retry handling. It does so without relying on any other external libraries beyond the Java standard library, and is compatible with Java 8 and up. So it will play nice with all of your projects, greenfield and legacy alike, without causing conflicts with any other dependency.

NOTE: Although the binary artifact produced by the project is backwards-compatible with Java 8, you do need JDK 9 or higher to modify or build the source code of this library itself.

This Change

Table of Contents
---

* [Installing the Driver](#installing-the-driver)
* [Initializing a Driver Instance](#initializing-a-driver-instance)
* [Key/Value Secret Engine Config](#key-value-secret-engine-config)
* [SSL Config](#ssl-config)

 * [General Options](#general-options)
 * [Java Keystore (JKS) based config](#java-keystore-jks-based-config)
 * [OpenSSL (PEM) based config](#openssl-pem-based-config)
* [Using the driver](#using-the-driver)
* [API Reference (Javadocs)](#api-reference-javadocs)
* [Version History](#version-history)
* [Development](#development)
* [License](#license)
* [Other Notes](#other-notes)

Installing the Driver

The driver is available from Maven Central, for all modern Java build systems.

Gradle:
```
dependencies {
  implementation 'com.bettercloud:vault-java-driver:5.1.0'
}
```

Maven:
```
<dependency>
  <groupId>com.bettercloud</groupId>
  <artifactId>vault-java-driver</artifactId>
  <version>5.1.0</version>
</dependency>
```

Initializing a Driver Instance

The `com.bettercloud.vault.VaultConfig` class is used to initialize a driver instance with desired settings.
In the most basic use cases, where you are only supplying a Vault server address and perhaps a root token, there are convenience constructor methods available:
```
final VaultConfig config = new VaultConfig()
                        .address("http://127.0.0.1:8200")
                        .token("3c9fd6be-7bc2-9d1f-6fb3-cd746c0fc4e8")
                        .build();

// You may choose not to provide a root token initially, if you plan to use
// the Vault driver to retrieve one programmatically from an auth backend.
final VaultConfig config = new VaultConfig().address("http://127.0.0.1:8200").build();
```

To explicitly set additional config parameters (*), you can use a builder pattern style to construct the `VaultConfig` instance. Either way, the initialization process will try to populate any unset values by looking to

environment variables.

```
final VaultConfig config =
  new VaultConfig()
      .address("http://127.0.0.1:8200")              // Defaults to "VAULT_ADDR" environment variable
      .token("3c9fd6be-7bc2-9d1f-6fb3-cd746c0fc4e8")  // Defaults to "VAULT_TOKEN" environment variable
      .openTimeout(5)                            // Defaults to "VAULT_OPEN_TIMEOUT" environment variable
      .readTimeout(30)                           // Defaults to "VAULT_READ_TIMEOUT" environment variable
      .sslConfig(new SslConfig().build())            // See "SSL Config" section below
      .build();
```

Once you have initialized a `VaultConfig` object, you can use it to construct an instance of the `Vault` primary driver class:

```
final Vault vault = new Vault(config);
```

Key Value Secret Engine Config

Shortly before its `1.0` release, Vault added a Version 2 of its [Key/Value Secrets Engine](https://www.vaultproject.io/docs/secrets/kv/index.html). This supports some addition features beyond the Version 1 that was the default in earlier Vault builds (e.g. secret rotation, soft deletes, etc).

Unfortunately, K/V V2 introduces some breaking changes, in terms of both request/response payloads as well as how URL's are constructed for Vault's REST API. Therefore, version `4.0.0` of this Vault Driver likewise had to introduce some breaking changes, to allow support for both K/V versions.

* **If you are using the new K/V V2 across the board**, then no action is needed. The Vault Driver now assumes this by default.

* **If you are still using the old K/V V1 across the board**, then you can use the `Vault` class constructor: `public Vault(final VaultConfig vaultConfig, final Integer engineVersion)`, supplying a `1` as the engine version parameter.
constructor, then you can declare whether to use Version 1 or 2 across the board.

* **If you are using a mix, of some secret paths mounted with V1 and others mounted with V2**, then you have two options:

* You can explicitly specify your Vault secret paths, and which K/V version each one is using. Construct your `Vault` objects with the constructor `public Vault(final VaultConfig vaultConfig, final Boolean useSecretsEnginePathMap, final Integer globalFallbackVersion)`.

Within the `VaultConfig` object, supply a map of Vault secret paths to their associated K/V version (`1` or `2`).

* You can rely on the Vault Driver to auto-detect your mounts and K/V versions upon instantiation. Use the same constructor as above,
but leave the map `null`. Note that this option requires your authentication credentials to have access to read Vault's `/v1/sys/mounts`
path.

Version 2 of the K/V engine dynamically injects a qualifier element into your secret paths, which varies depending on the type of for read and write operations, in between the root version
operation. For example, for read and write operations, the secret path:

```v1/mysecret```

... has a "data" qualifier injected:

```v1/data/mysecret```

The default behavior of this driver is to insert the appropriate qualifier one level deep (i.e. in between the root version number
and the rest of the path). However, if your secret path is prefixed, such that the qualifier should be injected further down:

```v1/my/long/prefix/data/anything/else```

... then you should accordingly set the `VaultConfig.prefixPathDepth` property when constructing your `Vault` instance.

SSL Config

If your Vault server uses a SSL certificate, then you must supply that certificate to establish connections. Also, if you are using certificate-based client authentication, then you must supply a client certificate and private key that have been previously registered with your Vault server.

SSL configuration has been broken off from the `VaultConfig` class, and placed in its own `SslConfig` class. This class likewise using a builder pattern.

General Options

```
.verify(false)    // Defaults to "VAULT_SSL_VERIFY" environment variable (or else "true")
```

To disable SSL certificate verification altogether, set `sslVerify(false)`. YOU SHOULD NOT DO THIS IS A REAL
PRODUCTION SETTING! However, it can be useful in a development or testing server context. If this value is explicitly set to `false`, then all other SSL config is basically unused.
```

Java Keystore (JKS) based config

You can provide the driver with a JKS truststore, containing Vault's server-side certificate for basic SSL, using one of the following three options:

`.trustStore(object)` - Supply an in-memory `java.security.KeyStore` file, containing Vault server cert(s) that can be trusted.

`.trustStoreFile(path)` - Same as the above, but the path references a JKS file on the filesystem.

`.trustStoreResource(path)` - Same as the above, but the path references a classpath resource rather than a filesystem path (e.g. if you've bundled the JKS file into your application's JAR, WAR, or EAR file).

If you are only using basic SSL, then no keystore need be provided. However, if you would like to use Vault's TLS Certificate auth backend for client side auth, then you need to provide a JKS keystore containing your client-side certificate and private key:

`.keyStore(object, password)` - Supply an in-memory `java.security.KeyStore` file containing a client certificate and private key, and the password needed to access it (can be null).
can be trusted.

`.keyStoreFile(path, password)` - Same as the above, but the path references a JKS file on the filesystem.

`.keyStoreResource(path, password)` - Same as the above, but the path references a classpath resource rather than a filesystem path (e.g. if you've bundled the JKS file into your application's JAR, WAR, or EAR file).

NOTE: JKS-based config trumps PEM-based config (see below). If for some reason you build an `SslConfig` object
with both JKS and PEM data present, then only the JKS data will be used. You cannot "mix-and-match", providing a JKS-based truststore and PEM-based client auth data.

OpenSSL (PEM) based config

To supply Vault's server-side certificate for basic SSL, you can use one of the following three options:

`.pemFile(path)` - Supply the path to an X.509 certificate in unencrypted PEM format, using UTF-8 encoding (defaults
to "VAULT_SSL_CERT" environment variable).

`.pemResource(path)` - Same as above, but the path references a classpath resource rather than a filesystem path (e.g. if
you've bundled the PEM file into your applications's JAR, WAR, or EAR file).

`.pemUTF8(contents)` - The string contents extracted from the PEM file. For Java to parse the certificate properly, there must be a line-break in between the certificate header and body (see the `SslConfig`

Javadocs for more detail).

If SSL verification is enabled, no JKS-based config is provided, AND none of these three methods are called, then `SslConfig` will by default check for a `VAULT_SSL_CERT` environment variable. If that's setw then it will be
treated as a filesystem path.

To use Vault's TLS Certificate auth backend for SSL client auth, you must provide your client certificate and private key, using some pair from the following options:

`.clientPemUTF8(path)` - Supply the path to an X.509 certificate in unencrypted PEM format, using UTF-8 encoding.

`.clientPemResource(path)` - Same as above, but the path references a classpath resource rather than a filesystem path (e.g. if
you've bundled the PEM file into your applications's JAR, WAR, or EAR file).

`.clientPemUTF8(contents)` - The string contents extracted from the PEM file. For Java to parse the certificate properly,
there must be a line-break in between the certificate header and body (see the `SslConfig` Javadocs for more detail).

`.clientKeyPemUTF8(path)` - Supply the path to an RSA private key in unencrypted PEM format, using UTF-8 encoding.

`.clientKeyPemResource(path)` - Same as above, but the path references a classpath resource rather than a filesystem path (e.g. if
you've bundled the PEM file into your applications's JAR, WAR, or EAR file).

`.clientKeyPemUTF8(contents)` - The string contents extracted from the PEM file. For Java to parse the certificate properly,
there must be a line-break in between the certificate header and body (see the `SslConfig` Javadocs for more detail).

Using the Driver

Like the `VaultConfig` class, `Vault` too supports a builder pattern DSL style:

```
final Map<String, String> secrets = new HashMap<String, String>();
secrets.put("value", "world");
secrets.put("other_value", "You can store multiple name/value pairs under a single key");

// Write operation
final LogicalResponse writeResponse = vault.logical()
                        .write("secret/hello", secrets);
```

...

// Read operation
final String value = vault.logical()
.read("secret/hello")
.getData().get("value");
```

`Vault` has a number of methods for accessing the classes that implement the various endpoints of Vault's HTTP API:

* `logical()`: Contains core operations such as reading and writing secrets.
* `auth()`: Exposes methods for working with Vault's various auth backends (e.g. to programmatically retrieve a token
by authenticating with a username and password).
* `pki()`: Operations on the PKI backend (e.g. create and delete roles, issue certificate credentials).
* `debug()`: Health check endpoints.

The driver DSL also allows you to specify retry logic, by chaining the `withRetries()` ahead of accessing the endpoint
implementation:

```
// Retry up to 5 times if failures occur, waiting 1000 milliseconds in between each retry attempt.
final LogicalResponse response = vault.withRetries(5, 1000)
.logical()
.read("secret/hello");
```

API Reference (Javadocs)

Full [Javadoc documentation](http://bettercloud.github.io/vault-java-driver/javadoc/).

Version History

Note that changes to the major version (i.e. the first number) represent possible breaking changes, and may require modifications in your code to migrate. Changes to the minor version (i.e. the second number) should represent non-breaking changes. The third number represents any very minor bugfix patches.

* **5.1.0**: This release contains the following updates:
* Supports path prefixes when using K/V engine V2. [(PR #189)](https://github.com/BetterCloud/vault-java-driver/pull/189)
* Fixes issues with bulk requests in the transit API. [(PR #195)](https://github.com/BetterCloud/vault-java-driver/pull/195)
* Adds response body to exception for Auth failures. [(PR #198)](https://github.com/BetterCloud/vault-java-driver/pull/198)
```

 * Support all options for the createToken operation. [(PR #199)](https://github.com/BetterCloud/vault-java-driver/pull/199)

* **5.0.0**: This release contains the following updates:
 * Changes the retry behavior, to no longer attempt retries on 4xx response codes (for which retries generally won't succeed anyway). This
   is the only (mildly) breaking change in this release, necessitating a major version bump. [(PR #176)](https://github.com/BetterCloud/vault-java-driver/pull/176)
 * Implements support for the Database secret engine. [(PR #175)](https://github.com/BetterCloud/vault-java-driver/pull/175)
 * Makes the "x-vault-token" header optional, to allow use of Vault Agent. [(PR #184)](https://github.com/BetterCloud/vault-java-driver/pull/184)
 * Removes stray uses of `System.out.println` in favor of `java.util.logging`. [(PR #178)](https://github.com/BetterCloud/vault-java-driver/pull/178)
 * Adds the enum constant `MountType.KEY_VALUE_V2`. [(PR #182)](https://github.com/BetterCloud/vault-java-driver/pull/182)

* **4.1.0**: This release contains the following updates:
 * Support for JWT authentication, for use by Kubernetes and other JWT-based authentication providers. [(PR #164)](https://github.com/BetterCloud/vault-java-driver/pull/164)
 * Updates the lease revoke method, to support changes in the underlying Vault API. [(PR #163)](https://github.com/BetterCloud/vault-java-driver/pull/163)
 * Changes the `VaultConfig.secretsEnginePathMap(...)` method from default access level to `public`, to allow for manual
   setting [(PR #164)](https://github.com/BetterCloud/vault-java-driver/pull/156)
 * Adds the nonce value to `AuthResponse`, to facilitate re-authentication with Vault via AWS. [(PR #168)](https://github.com/BetterCloud/vault-java-driver/pull/168)
 * Establishes a `module-info` file, updates the JDK requirement for building this library to Java 9 (although the built
   library artifact remains compatible as a dependency in Java 8 projects). [(PR #165)](https://github.com/BetterCloud/vault-java-driver/pull/165)
 * Updates Gradle, and various test dependencies to their latest versions. Integration tests now target Vault 1.1.3.

* **4.0.0**: This is a breaking-change release, with two primary updates:
 * Adds support for Version 2 of the Key/Value Secrets Engine. The driver now assumes that your Vault instance uses Version 2 of the
   Key/Value Secrets Engine across the board. To configure this, see the [Key/Value Secret Engine Config](#key-value-secret-engine-config)
   section above.
 * Adds support for the namespaces feature of Vault Enterprise.

* **3.1.0**: Several updates.
 * Adds support for seal-related operations (i.e. `/sys/seal`, `/sys/unseal`, `/sys/seal-status`).
 * Adds support for the AWS auth backend.
 * Adds support for the Google Cloud Platform auth backend.
 * Adds support for the LDAP auth Backend.
 * Allows auth backend methods to be configured for non-default mount points.
 * Adds "revoke-self" capability for auth tokens.

* Adds support for response-wrapping token validation
* Support for signing a new certificate based on a CSR (i.e. the `/v1/pki/sign` endpoint).
* Support for the PKI backend revoke method, and addition of a useCsrSans property in PKI role object
* Gives `VaultConfig` the ability to disable loading from environment variables if desired.
* Cleans up issues with the new Docker-based integration test suite.
* Updates all dependencies to their latest versions (including switching to Vault 0.9.1 for integration testing).

* **3.0.0**: This is a breaking-change release, with several updates.
* **API changes**:
  * Adds support for writing arbitrary objects to Vault, instead of just strings (i.e. the `com.bettercloud.vault.api.Logical.write(...)` method now accepts a `Map<String. Object>` rather than a `Map<String, String>`).
  * Refactors the `VaultConfig` class, forcing use of the builder pattern and breaking off SSL-related config into a separate `SslConfig` class.
  * Refactors the `Auth.createToken()` method, to encapsulate the possible options within a config object rather than having the method signature contain 8 optional arguments.
* **SSL and Auth Backend support**:
  * Adds support for authenticating with the TLS Certificate auth backend.
  * Updates SSL support in general, allowing users to configure the driver with Java-friendly JKS keystore and truststore files (in addition to continuing to support Vault-friendly PEM format).
  * Implements the `/v1/auth/token/lookup-self` endpoint.
  * Supports creating tokens against a role.
* **Major re-vamp of the integration test suite**:
  * The tests now use the [TestContainers](https://www.testcontainers.org/) library to setup and launch a Vault server instance from within a Docker container, in a completely automated manner. You no longer have to

    manually configure and run a Vault server to use the test suite!
  * The tests are now going against a regular Vault server, rather than one running in "dev mode". Therefore, they are now able to use HTTPS connections rather than plain HTTP.
  * Upgrades tests to use Java 8 (although the library itself still targets Java 7).
* **Misc / quality-of-life**:
  * Includes the REST response body in `VaultException` messages for basic read and write operations.
  * Makes numerous classes implement `Serializable`.
  * Upgrades the project to Gradle 4.0.

* **2.0.0**: This is breaking-change release, with numerous deprecated items cleaned up.
* Adds support for authentication via the AppRole auth backend.
* Adds support for renewing secret leases.
* Removes the `com.bettercloud.vault.api.Sys` class, deprecated in the 1.2.0 release.
* Removes the `com.bettercloud.vault.api.Auth.loginByUsernamePassword` method, deprecated in the 1.2.0 release.
* Removes the fields `leaseId`, `leaseDuration`, and `renewable` from the `VaultResponse` base class, instead including them only in the subclasses for specific response types where they are found.
* Changes the `com.bettercloud.vault.response.AuthReponse` class field `authLeaseDuration` from type `int` to `long`.
* Refactors and removes various deprecated `private` methods, with no change to the exposed API.

* **1.2.0**: This is a substantial release, with numerous additions. It's a minor version number only because there

should be no breaking changes. The changes include the following:
 * Switches from Vault 0.5.x to 0.6.x for automated tests.
 * Adds a field to `VaultException` for capturing the HTTP response code (if any) from Vault.
 * Updates the Gradle build, so that you no longer need empty placeholder values for certain variables elsewhere in your environment.
 * Updates integration test suite to account for breaking changes in Vault 0.6.x (e.g. you can no longer use a token that was obtained from one of the authentication backends to perform tasks such as creating and deleting roles, etc).
 * Deprecates the App ID authentication backend, and adds a new version of the Userpass authentication backend that
doesn't require a path prefix. Adds support for the GitHub authentication backend.
 * If the `VAULT_TOKEN` environment parameter is not set, then the driver will now check for a file named `.vault-token`
in the executing user's home directory, and try to read a token value from that.
 * Deprecates the `com.bettercloud.vault.api.Sys` class, moving the debug-related methods into their own specific `com.bettercloud.vault.api.Debug` class instead.
 * Implements some of the lease related endpoints (i.e. revoke, revoke-prefix, revoke-force).
 * Supports PKI backends that are mounted on non-default paths.

* **1.1.1**: Changes the `ttl` argument to `Pki.issue()` from `Integer` to `String`, to fix a bug preventing you from specifying the time suffix (e.g. "1h").
* **1.1.0**: Switches from Vault 0.4.x to 0.5.x for automated tests. Adds support to the Logical API wrapper for listing and deleting secrets. Implements the `/v1/sys/health` health-check HTTP API endpoint. Implements portions of the PKI backend (e.g. creating and deleting roles, issuing credentials). Marks numerous methods as deprecated, to be removed in a future major release.
* **1.0.0**: Drops support for Java 6. Removes all methods marked as `@Deprecated` in version 0.5.0. Adds support for response metadata (i.e. "lease_id", "renewable", "lease_duration") to all response types, rather than just `AuthResponse`. Changes `leaseDuration` type from `int` to `Long` in `AuthResponse`. Removes `final` declarations on all classes (outside of the JSON package). Various bugfixes. Adds support for auth token self-renewal. Adds support for writing values that return content.
* **0.5.0**: Adds support for supplying SSL certificates, and for toggling whether or not the Vault server's SSL certificate will be verified. Also adds support for "openTimeout" and "readTimeout" settings. Deprecates the "timeout", "sslTimeout", "proxyAddress", "proxyPort", "proxyUsername", and "proxyPassword" settings (the proxy settings may return in a future version, but it's too misleading to have methods exposed for settings that won't really be supported soon).
* **0.3.0**: Initial public release. Support for writing and reading secrets, authenticating with the "AppID" or "Username & Password" auth backends. All over-the-wire methods support automatic retry logic.

Development

Pull requests are welcomed for bugfixes or enhancements that do not alter the external facing class and method signatures. For any breaking changes that would alter the contract provided by this driver, please open up an issue to discuss it first.

All code changes should include unit test and/or integration test coverage as appropriate. Unit tests are any that can be run in isolation, with no external dependencies. Integration tests are those which require a Vault server instance (at least a Dev Server) up and running.

Unit tests are located under the `src/test` directory, and can be run with the Grade `unitTest` task.

Integration tests are located under the `src/test-integration` directory, and can be run with the Gradle `integrationTest` task. See the additional `README.md` file in this directory for more detailed information.

Although this library now includes a `module-info` class for use by Java 9+, the library currently targets Java 8 compatibility. Please do not attempt to introduce any features or syntax not compatible with Java 8 (the Gradle build script should prevent you from doing so without modification).

License

The MIT License (MIT)

Copyright (c) 2016-2019 BetterCloud

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Other Notes

The Vault server system itself is a product of HashiCorp, a completely separate organization.

This client driver adapts JSON parsing code from Ralf Sternberg's excellent [minimal-json](https://github.com/ralfstx/minimal-json) library, likewise available under the MIT License. Package names have all been changed, to prevent any conflicts should you happen to be using a different version of that library elsewhere in your project dependencies.

Found in path(s):
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/README.md
No license file was found, but licenses were detected in source scan.

/**

```
* <p>Intended for internal use by other classes within the Vault driver library.</p>
*
* <p>JSON parsing functionality.  The classes in this package are adapted from Ralf Sternberg's excellent
* <a href="https://github.com/ralfstx/minimal-json">"minimal-json"</a> library, available under the MIT License.
* The package names have all been changed, to prevent any conflicts should you happen to be using a different
* version of this library elsewhere in your project dependencies.</p>
*/
```

Found in path(s):
* /opt/cola/permits/1162863465_1653505868.969204/0/vault-java-driver-master-1-zip/vault-java-driver-master/src/main/java/com/bettercloud/vault/json/package-info.java

1.130 okio 2.8.0

1.130.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2015 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/SegmentedByteString.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/SegmentedByteString.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/ForwardingTimeout.kt
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
```

* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/internal/RealBufferedSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/internal/RealBufferedSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/internal/SegmentedByteString.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2017 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Utf8.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/internal/Buffer.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Timeout.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/BufferedSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Sink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Buffer.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/RealBufferedSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/BufferedSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Source.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-
jar/commonMain/okio/RealBufferedSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Okio.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/SegmentPool.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/InflaterSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/ForwardingSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/BufferedSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/JvmOkio.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Segment.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/BufferedSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/GzipSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/DeflaterSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/GzipSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Buffer.kt

* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Source.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Sink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Timeout.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/RealBufferedSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/SegmentPool.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/RealBufferedSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/AsyncTimeout.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/ForwardingSink.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/-Util.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/internal/ByteString.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/-Platform.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/-Platform.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/internal/-Utf8.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/ByteString.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 Square Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and

* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/ByteString.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/-DeprecatedUpgrade.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/-DeprecatedUtf8.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/PeekSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/-DeprecatedOkio.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Throttler.kt
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/-Base64.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/HashingSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Pipe.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/HashingSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Options.kt

1.131 okio 3.0.0

1.131.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2017 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/Utf8.kt
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2020 Square, Inc. and others.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/CipherSink.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/CipherSource.kt
No license file was found, but licenses were detected in source scan.

```
/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements.  See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License.  You may obtain a copy of the License at
*
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/internal/ZipEntry.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/internal/zip.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/ZipFileSystem.kt

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/internal/-RealBufferedSource.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/internal/-SegmentedByteString.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/internal/-RealBufferedSink.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/Source.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/BufferedSink.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-

jar/commonMain/okio/RealBufferedSource.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/internal/-Buffer.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/Sink.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/Okio.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/RealBufferedSink.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/Timeout.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/Buffer.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/BufferedSource.kt
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2016 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/HashingSource.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/Pipe.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/Options.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/HashingSink.kt
No license file was found, but licenses were detected in source scan.

```
/*
*  Licensed to the Apache Software Foundation (ASF) under one or more
*  contributor license agreements.  See the NOTICE file distributed with
*  this work for additional information regarding copyright ownership.
*  The ASF licenses this file to You under the Apache License, Version 2.0
*  (the "License"); you may not use this file except in compliance with
*  the License.  You may obtain a copy of the License at
*
*     http://www.apache.org/licenses/LICENSE-2.0
*
*  Unless required by applicable law or agreed to in writing, software
*  distributed under the License is distributed on an "AS IS" BASIS,
```

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/-Base64.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2021 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/internal/ResourceFileSystem.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/FileSystem.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/FileHandle.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/internal/-FileSystem.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/internal/FixedLengthSource.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/JvmFileHandle.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2015 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/SegmentedByteString.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/ForwardingTimeout.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/SegmentedByteString.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/ByteString.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/internal/-ByteString.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/-JvmPlatform.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/internal/-Utf8.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/-Util.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/-CommonPlatform.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/RealBufferedSource.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/AsyncTimeout.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/Buffer.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/ForwardingSource.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/SegmentPool.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/BufferedSink.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/InflaterSource.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/SegmentPool.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/Timeout.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/GzipSource.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/DeflaterSink.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/Segment.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/RealBufferedSink.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/JvmOkio.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/BufferedSource.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/ForwardingSink.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/GzipSink.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 Square Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/ByteString.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2020 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/NioSystemFileSystem.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/HashingSource.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/internal/-Path.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/Path.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/Path.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/ForwardingFileSystem.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/HashingSink.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/FileSystem.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/ExperimentalFileSystem.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/JvmSystemFileSystem.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/FileMetadata.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/-DeprecatedUtf8.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/Throttler.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/-DeprecatedUpgrade.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/jvmMain/okio/-DeprecatedOkio.kt
* /opt/cola/permits/1519136854_1671710664.1790016/0/okio-3-0-0-sources-3-jar/commonMain/okio/PeekSource.kt

1.132 spring-commons-logging-bridge 5.3.22

1.132.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Spring Framework ${version}
Copyright (c) 2002-${copyright} Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for
these subcomponents is subject to the terms and conditions of the
subcomponent's license, as noted in the license.txt file.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including

the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each

Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

==========

SPRING FRAMEWORK ${version} SUBCOMPONENTS:

Spring Framework ${version} includes a number of subcomponents
with separate copyright notices and license terms. The product that
includes this file does not necessarily use all the open source
subcomponents referred to below. Your use of the source
code for these subcomponents is subject to the terms and
conditions of the following licenses.

>>> ASM 9.1 (org.ow2.asm:asm:9.1, org.ow2.asm:asm-commons:9.1):

Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF

SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1999-2009, OW2 Consortium <https://www.ow2.org/>

>>> CGLIB 3.3 (cglib:cglib:3.3):

Per the LICENSE file in the CGLIB JAR distribution downloaded from https://github.com/cglib/cglib/releases/download/RELEASE_3_3_0/cglib-3.3.0.jar, CGLIB 3.3 is licensed under the Apache License, version 2.0, the text of which is included above.

>>> Objenesis 3.2 (org.objenesis:objenesis:3.2):

Per the LICENSE file in the Objenesis ZIP distribution downloaded from http://objenesis.org/download.html, Objenesis 3.2 is licensed under the Apache License, version 2.0, the text of which is included above.

Per the NOTICE file in the Objenesis ZIP distribution downloaded from http://objenesis.org/download.html and corresponding to section 4d of the Apache License, Version 2.0, in this case for Objenesis:

Objenesis
Copyright 2006-2019 Joe Walnes, Henri Tremblay, Leonardo Mesquita

==========

To the extent any open source components are licensed under the EPL and/or other similar licenses that require the source code and/or modifications to source code to be made available (as would be noted above), you may obtain a copy of the source code corresponding to the binaries for such open source components and modifications thereto, if any, (the "Source Files"), by downloading the Source Files from https://spring.io/projects, Pivotal's website at https://network.pivotal.io/open-source, or by sending a request, with your name and address to: Pivotal Software, Inc., 875 Howard Street, 5th floor, San Francisco, CA 94103, Attention: General Counsel. All such requests should clearly specify: OPEN SOURCE FILES REQUEST, Attention General Counsel. Pivotal can mail a copy of the Source Files to you on a CD or equivalent physical medium.

This offer to obtain a copy of the Source Files is valid for three years from the date you acquired this Software product. Alternatively, the Source Files

may accompany the Software.

1.133 duct-tape 1.0.8

1.133.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Richard North

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.134 asm-analysis 9.2

1.134.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//    notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//    notice, this list of conditions and the following disclaimer in the
//    documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//    this software without specific prior written permission.
/**
  * Constructs a new {@link SourceInterpreter} for the latest ASM API version. <i>Subclasses must
  * not use this constructor</i>. Instead, they must use the {@link #SourceInterpreter(int)}
```

* version.
*/

Found in path(s):
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/SourceInterpreter.java
No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*   contributors may be used to endorse or promote products derived from
*   this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/package.html
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//   notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//   notice, this list of conditions and the following disclaimer in the

// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/SimpleVerifier.java
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/BasicValue.java
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/AnalyzerException.java
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/Subroutine.java
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/Frame.java
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/BasicVerifier.java
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/SourceValue.java
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/Interpreter.java
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/Analyzer.java
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/Value.java
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-jar/org/objectweb/asm/tree/analysis/SmallSet.java
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
```
/**
 * Constructs a new {@link BasicInterpreter} for the latest ASM API version. <i>Subclasses must
 * not use this constructor</i>. Instead, they must use the {@link #BasicInterpreter(int)}
 * version.
 */
```

Found in path(s):
* /opt/cola/permits/1288694723_1647366364.82/0/asm-analysis-9-2-sources-

jar/org/objectweb/asm/tree/analysis/BasicInterpreter.java

1.135 findbugs-jsr305 3.0.2

1.135.1 Available under license :

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

1. "Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.
2. "Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.
3. "Licensor" means the individual or entity that offers the Work under the terms of this License.
4. "Original Author" means the individual or entity who created the Work.
5. "Work" means the copyrightable work of authorship offered under the terms of this License.
6. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

1. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
2. to create and reproduce Derivative Works;

3. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
4. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.
5.

For the avoidance of doubt, where the work is a musical composition:

1. Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

2. Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

6. Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions.The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

1. You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.
2. If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author,"

or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTIBILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

1. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
2. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

1. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

2. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
3. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
4. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
5. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There

are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

1.136 apache-kafka 3.1.1

1.136.1 Available under license :

Apache Kafka
Copyright 2022 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

This distribution has a binary dependency on jersey, which is available under the CDDL License. The source code of jersey can be found at https://github.com/jersey/jersey/.

This distribution has a binary test dependency on jqwik, which is available under the Eclipse Public License 2.0. The source code can be found at https://github.com/jlink/jqwik.

The streams-scala (streams/streams-scala) module was donated by Lightbend and the original code was copyrighted by them:
Copyright (C) 2018 Lightbend Inc. <https://www.lightbend.com>
Copyright (C) 2017-2018 Alexis Seigneurin.

This project contains the following code copied from Apache Hadoop:
clients/src/main/java/org/apache/kafka/common/utils/PureJavaCrc32C.java
Some portions of this file Copyright (c) 2004-2006 Intel Corporation and licensed under the BSD license.

This project contains the following code copied from Apache Hive:
streams/src/main/java/org/apache/kafka/streams/state/internals/Murmur3.java

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.137 jacoco 0.8.8

1.137.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--
  Copyright (c) 2009, 2022 Mountainminds GmbH & Co. KG and Contributors
  This program and the accompanying materials are made available under
  the terms of the Eclipse Public License 2.0 which is available at
  http://www.eclipse.org/legal/epl-2.0

  SPDX-License-Identifier: EPL-2.0

  Contributors:
    Evgeny Mandrikov - initial API and implementation
-->

Found in path(s):
* /opt/cola/permits/1324812216_1668300253.5014505/0/jacoco-maven-plugin-0-8-8-1-jar/META-INF/m2e/lifecycle-mapping-metadata.xml
* /opt/cola/permits/1324812216_1668300253.5014505/0/jacoco-maven-plugin-0-8-8-1-jar/META-INF/maven/org.jacoco/jacoco-maven-plugin/pom.xml

1.138 picocli 4.6.1

1.138.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
  Copyright 2017 Remko Popma

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

     http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
*/
/**
     * Called when either the command, option or positional parameter that has this preprocessor configured was
     * recognized by the picocli parser.

```
     * <p>Implementors are free to modify one or more of the specified command line arguments before they are
     * processed by the picocli parser (or by the option's {@link IParameterConsumer parameter consumer}, if one is specified).
     * </p><p>
     * Implementors may optionally <em>consume</em> one or more of the specified command line arguments:
     * a return value of {@code true} signals that the preprocessor consumed the parameter:
     * picocli should skip further processing of this option or positional parameter,
     * and the preprocessor implementation takes responsibility for assigning the
     * option or positional parameter a value. A return value of {@code false} means that picocli should
     * process the stack as usual for this option or positional parameter, and picocli is responsible for
     * assigning this option or positional parameter a value.
     * </p><p>
     * For a command, returning {@code true} signals that the preprocessor takes responsibility for parsing all
     * options and positional parameters for this command, and takes responsibility for validating constraints like
     * whether all required options and positional parameters were specified.
     * A return value of {@code false} means that picocli should
     * process the stack as usual for this command, and picocli is responsible for validation.
     * Command preprocessors can signal back to the picocli parser when they detect that the user requested version
     * information or usage help by putting a value of {@code true} in the specified
     * {@code info} map for keys {@code versionHelpRequested} or {@code usageHelpRequested}, respectively.
     * </p><p>
     * If the user input is invalid, implementations should throw a {@link ParameterException}
     * with a message to display to the user.
     * </p>
     * @param args the remaining command line arguments that follow the matched argument
     *          (the matched argument is not on the stack anymore)
     * @param commandSpec the command or subcommand that was matched (if the specified {@code argSpec} is
     *          {@code null}), or the command that the matched option or positional parameter belongs to
     * @param argSpec the option or positional parameter for which to pre-process command line arguments
     *          (may be {@code null} when this method is called for a subcommand that was matched)
     * @param info a map containing additional information on the current parser state,
     *              including whether the option parameter was attached to the option name with
     *              a `=` separator, whether quotes have already been stripped off the option, etc.
     *          Implementations may modify this map to communicate back to the picocli parser.
     *          Supported values:
     *          <table>
     *            <tr>
     *              <th>key</th><th>valid values</th><th>type</th>
     *            </tr>
     *            <tr>
     *              <td>separator</td><td>" (empty string): attached without separator, ' ' (space): not attached, (any other string): option name was attached to option param with specified separator</td>
     *              <td>java.lang.String</td>
     *            </tr>
     *            <tr>
     *              <td>negated</td>
     *              <td>{@code true}: the option or positional parameter is a {@linkplain Option#negatable() negated} option/parameter, {@code false}: the option or positional parameter is not a negated
```

option/parameter</td>
     *   <td>java.lang.Boolean</td>
     *   </tr>
     *   <tr>
     *   <td>unquoted</td>
     *   <td>{@code true}: quotes surrounding the value have already been stripped off, {@code false}:
quotes surrounding the value have not yet been stripped off</td>
     *   <td>java.lang.Boolean</td>
     *   </tr>
     *   <tr>
     *   <td>versionHelpRequested</td>
     *   <td>{@code true}: version help was requested, {@code false}: version help was not
requested</td>
     *   <td>java.lang.Boolean</td>
     *   </tr>
     *   <tr>
     *   <td>usageHelpRequested</td>
     *   <td>{@code true}: usage help was requested, {@code false}: usage help was not requested</td>
     *   <td>java.lang.Boolean</td>
     *   </tr>
     *   </table>
     * @returns true if the preprocessor consumed the parameter
     *   and picocli should skip further processing of the stack for this option or positional parameter;
     *   false if picocli should continue processing the stack for this option or positional parameter
     * @throws ParameterException if the user input is invalid
     */

Found in path(s):
* /opt/cola/permits/1178487930_1625660844.82/0/picocli-4-6-1-sources-1-jar/picocli/CommandLine.java
No license file was found, but licenses were detected in source scan.

/*
  Copyright 2017 Remko Popma

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

     http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1178487930_1625660844.82/0/picocli-4-6-1-sources-1-jar/picocli/AutoComplete.java

1.139 junit-jupiter-aggregator 5.8.2

1.139.1 Available under license :

Eclipse Public License - v 2.0
==========

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:
* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **b)** in the case of each subsequent Contributor:
 * **i)** changes to the Program, and
 * **ii)** additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

3.1 If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
 * **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
* **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
* **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
* **b)** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the

appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
Open Source Licenses
==========

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for
these subcomponents is subject to the terms and conditions of the
subcomponent's license, as noted in the LICENSE-<subcomponent>.md
files.

1.140 jackson-databind 2.13.3

1.140.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0
To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.141 brave-propagation:-amazon-web-services-(aws) 0.21.3

1.141.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2018 The OpenZipkin Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1340816205_1654861315.2555492/0/brave-propagation-aws-0-21-3-sources-jar/brave/propagation/aws/HexCodec.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2020 The OpenZipkin Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1340816205_1654861315.2555492/0/brave-propagation-aws-0-21-3-sources-jar/brave/propagation/aws/AWSPropagation.java
* /opt/cola/permits/1340816205_1654861315.2555492/0/brave-propagation-aws-0-21-3-sources-jar/brave/propagation/aws/AWSExtractor.java
* /opt/cola/permits/1340816205_1654861315.2555492/0/brave-propagation-aws-0-21-3-sources-jar/brave/propagation/aws/AWSInjector.java
No license file was found, but licenses were detected in source scan.

<!--

Copyright 2016-2020 The OpenZipkin Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340816205_1654861315.2555492/0/brave-propagation-aws-0-21-3-sources-jar/META-INF/maven/io.zipkin.aws/brave-propagation-aws/pom.xml

1.142 junit-platform-junit-platform-engine 1.8.2

1.142.1 Available under license :

Eclipse Public License - v 2.0
==========

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:
* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and

* **b)** in the case of each subsequent Contributor:
 * **i)** changes to the Program, and
 * **ii)** additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license

shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

3.1 If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
 * **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 * **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 * **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
 * **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
* **b)** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
Open Source Licenses
==========

This product may include a number of subcomponents with separate

copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.143 brave-instrumentation:-mongodb 5.13.9

1.143.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0
* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed
by Square, Inc in the Moshi Library:

* Copyright (C) 2010 Google Inc.
* License: Apache License v2.0
* Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

* License: Apache License v2.0
* Homepage: https://github.com/raphw/weak-lock-free

1.144 netty/resolver/dns/classes/macos 4.1.79.Final

1.144.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2021 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~   https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->

Found in path(s):
* /opt/cola/permits/1519137104_1671710620.1352448/0/netty-resolver-dns-classes-macos-4-1-79-final-sources-3-jar/META-INF/maven/io.netty/netty-resolver-dns-classes-macos/pom.xml
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137104_1671710620.1352448/0/netty-resolver-dns-classes-macos-4-1-79-final-sources-3-jar/io/netty/resolver/dns/macos/DnsResolver.java
* /opt/cola/permits/1519137104_1671710620.1352448/0/netty-resolver-dns-classes-macos-4-1-79-final-sources-3-jar/io/netty/resolver/dns/macos/MacOSDnsServerAddressStreamProvider.java
* /opt/cola/permits/1519137104_1671710620.1352448/0/netty-resolver-dns-classes-macos-4-1-79-final-sources-3-jar/io/netty/resolver/dns/macos/package-info.java

1.145 java-classmate classmate-1.5.1

1.145.1 Available under license :

Java ClassMate library was originally written by Tatu Saloranta (tatu.saloranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel

This copy of Java ClassMate library is licensed under Apache (Software) License,
version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but

not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their

Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with

the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.146 apache-log4j-layout-for-json-template 2.17.2

1.146.1 Available under license :

Apache Log4j Layout for JSON template
Copyright 1999-2022 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and

limitations under the License.

1.147 spring-boot-actuator-autoconfigure 2.7.2

1.147.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Spring Boot 2.7.2
Copyright (c) 2012-2022 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

1.148 spring-framework 5.3.22

1.148.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the

direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and

wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor

has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Spring Framework ${version}
Copyright (c) 2002-${copyright} Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for

these subcomponents is subject to the terms and conditions of the
subcomponent's license, as noted in the license.txt file.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

==========

SPRING FRAMEWORK ${version} SUBCOMPONENTS:

Spring Framework ${version} includes a number of subcomponents
with separate copyright notices and license terms. The product that
includes this file does not necessarily use all the open source
subcomponents referred to below. Your use of the source
code for these subcomponents is subject to the terms and
conditions of the following licenses.

>>> ASM 9.1 (org.ow2.asm:asm:9.1, org.ow2.asm:asm-commons:9.1):

Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE

LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1999-2009, OW2 Consortium <https://www.ow2.org/>

>>> CGLIB 3.3 (cglib:cglib:3.3):

Per the LICENSE file in the CGLIB JAR distribution downloaded from https://github.com/cglib/cglib/releases/download/RELEASE_3_3_0/cglib-3.3.0.jar, CGLIB 3.3 is licensed under the Apache License, version 2.0, the text of which is included above.

>>> Objenesis 3.2 (org.objenesis:objenesis:3.2):

Per the LICENSE file in the Objenesis ZIP distribution downloaded from http://objenesis.org/download.html, Objenesis 3.2 is licensed under the Apache License, version 2.0, the text of which is included above.

Per the NOTICE file in the Objenesis ZIP distribution downloaded from http://objenesis.org/download.html and corresponding to section 4d of the Apache License, Version 2.0, in this case for Objenesis:

Objenesis
Copyright 2006-2019 Joe Walnes, Henri Tremblay, Leonardo Mesquita

==========

To the extent any open source components are licensed under the EPL and/or other similar licenses that require the source code and/or modifications to source code to be made available (as would be noted above), you may obtain a copy of the source code corresponding to the binaries for such open source components and modifications thereto, if any, (the "Source Files"), by downloading the Source Files from https://spring.io/projects, Pivotal's website at https://network.pivotal.io/open-source, or by sending a request, with your name and address to: Pivotal Software, Inc., 875 Howard Street, 5th floor, San Francisco, CA 94103, Attention: General Counsel. All such requests should clearly specify: OPEN SOURCE FILES REQUEST, Attention General Counsel. Pivotal can mail a copy of the Source Files to you on a CD or equivalent physical medium.

This offer to obtain a copy of the Source Files is valid for three years from the date you acquired this Software product. Alternatively, the Source Files may accompany the Software.

1.149 checker-qual 3.5.0

1.149.1 Available under license :

Checker Framework qualifiers
Copyright 2004-present by the Checker Framework developers

MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.150 docker-java-transport-zerodep 3.2.13

1.150.1 Available under license :

Apache Commons Codec
Copyright 2002-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from http://aspell.net/test/orig/batch0.tab.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

==========

The content of package org.apache.commons.codec.language.bm has been translated

from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

Apache HttpClient
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work

(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses

granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"

replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.151 jackson-module-parameter-names 2.13.3

1.151.1 Available under license :

Apache-2.0

1.152 hamcrest v2.2

1.152.1 Available under license :

BSD License

Copyright (c) 2000-2015 www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written

permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.153 open-telemetry/opentelemetry-java 1.14.0

1.153.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The gRPC Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1343419668_1655584013.3890648/0/opentelemetry-exporter-otlp-common-1-14-0-sources-jar/io/opentelemetry/exporter/internal/grpc/OkHttpGrpcExporter.java
* /opt/cola/permits/1343419668_1655584013.3890648/0/opentelemetry-exporter-otlp-common-1-14-0-sources-jar/io/opentelemetry/exporter/internal/grpc/MarshalerInputStream.java

No license file was found, but licenses were detected in source scan.

// Copyright 2008 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
//   * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
//   * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
//   * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1343419668_1655584013.3890648/0/opentelemetry-exporter-otlp-common-1-14-0-sources-jar/io/opentelemetry/exporter/internal/marshal/CodedOutputStream.java
* /opt/cola/permits/1343419668_1655584013.3890648/0/opentelemetry-exporter-otlp-common-1-14-0-sources-jar/io/opentelemetry/exporter/internal/marshal/CodedInputStream.java
* /opt/cola/permits/1343419668_1655584013.3890648/0/opentelemetry-exporter-otlp-common-1-14-0-sources-jar/io/opentelemetry/exporter/internal/marshal/WireFormat.java

1.154 bean-validation-api 2.0.2

1.154.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity

exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 2013 Cognifide Limited

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.155 fabric8-::-kubernetes-model-::-apps 5.12.1

1.155.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

Copyright (C) 2015 Red Hat, Inc.

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340816142_1654860957.9552085/0/kubernetes-model-apps-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-apps/pom.xml

1.156 aspectj-weaver 1.9.7

1.156.1 Available under license :

No license file was found, but licenses were detected in source scan.

/* ==
* The Apache Software License, Version 1.1
*
* Copyright (c) 2001 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in
*   the documentation and/or other materials provided with the
*   distribution.
*
* 3. The end-user documentation included with the redistribution,
*   if any, must include the following acknowledgment:
*   "This product includes software developed by the
*   Apache Software Foundation (http://www.apache.org/)."
*   Alternately, this acknowledgment may appear in the software itself,
*   if and wherever such third-party acknowledgments normally appear.
*

```
* 4. The names "Apache" and "Apache Software Foundation" and
*    "Apache BCEL" must not be used to endorse or promote products
*    derived from this software without prior written permission. For
*    written permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache",
*    "Apache BCEL", nor may "Apache" appear in their name, without
*    prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED.  IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation.  For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.
*
* Extended by Adrian Colyer, June 2005 to support unpacking of Signature
* attribute
*/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-
jar/org/aspectj/apache/bcel/classfile/Signature.java
No license file was found, but licenses were detected in source scan.

/* ********************************************************************
* Copyright (c) 2004 IBM
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Andy Clement -     initial implementation
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/NameValuePair.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Wes Isberg       initial implementation
*   Andy Clement       fleshed out to match SuppressWarnings
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/SuppressAjWarnings.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
* Copyright (c) 2013 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* initial implementation              Andy Clement
*******************************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/control/CodeGenerationHint.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
```

*
* Contributors:
*   Wes Isberg   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/internal/lang/PlaceHolder.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002 Palo Alto Research Center, Incorporated (PARC).
*   2005,2020 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/World.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005-2010 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/TypeVariable.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/TypeFactory.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2004 IBM Corporation and others.
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*

* Contributors:
*   IBM Corporation - initial API and implementation
**********/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/HasAnnotation.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/StandardPointcutParser.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/IClassFileProvider.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/IWeaveRequestor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/UnwovenClassFileWithThirdPartyManagedBytecode.java
No license file was found, but licenses were detected in source scan.

/* **********
* Copyright (c) 2008 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors
* Andy Clement - extracted from AnnotationTypePattern
* **********/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AnyAnnotationTypePattern.java
No license file was found, but licenses were detected in source scan.

/* **********
* Copyright (c) 2011 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement - SpringSource/vmware
* **********/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/bcel/AspectInstanceVar.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Xerox/PARC   initial implementation
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/Reflection.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/TypeSafeEnum.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Adrian Colyer   Initial implementation
*   Nieraj Singh
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/HasMemberTypePattern.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2002 Palo Alto Research Center, Incorporated (PARC).
*   2005 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt

*
* Contributors:
*   PARC   initial implementation
*   AMC   extracted as interface
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ResolvedMember.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* initial implementation   Alexandre Vasseur
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/AfterReturning.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/After.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclarePrecedence.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/Pointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/AfterThrowing.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/AtAjAttributes.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/Aspect.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/Before.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/Around.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelPerClauseAspectAdder.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/ProceedingJoinPoint.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2012 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Lyor Goldstein (vmware) add support for weaved class being re-defined
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/AsynchronousFileCacheBacking.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/ZippedFileCacheBacking.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2002 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   PARC   initial implementation
*   Andy Clement   pushed down into bcel module
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/LineNumberTag.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   PARC   initial implementation
*   Alexandre Vasseur   @AspectJ ITDs

* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ResolvedType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ResolvedTypeMunger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelTypeMunger.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2007 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement, IBM   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ExposeTypeMunger.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2003,2010 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Mik Kersten   initial implementation
*   Andy Clement, IBM, SpringSource   Extensions for better IDE representation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/internal/ProgramElement.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2011 Contributors.

* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Abraham Nevado - Lucierna initial implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/definition/LightXMLParser.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2006 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* variant of Aspects in the aspectj5rt project - this one isn't Java5   - Andy Clement
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/Aspects14.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2006 IBM Corporation and others.
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Matthew Webster - initial implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/Traceable.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/ISupportsMessageContext.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/Trace.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/AbstractTrace.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/DefaultTrace.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/TraceFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/DefaultTraceFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/Constants.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/IMessageContext.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/Jdk14Trace.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/CommonsTraceFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/Jdk14TraceFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/CommonsTrace.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2002,2010
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     PARC     initial implementation
*     Adrian Colyer, IBM
*     Andy Clement, IBM, SpringSource
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PatternParser.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2008 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
```

* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/StandardAnnotation.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelAnnotation.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/TemporaryTypeMunger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/IUnwovenClassFile.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AbstractAnnotationAJ.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 1999-2001 Xerox Corporation,
*   2002 Palo Alto Research Center, Incorporated (PARC),
*   2004 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Xerox/PARC   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/SoftException.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2008, 2018 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Andy Clement
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/asm/StackMapAdder.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2009 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     initial implementation   Andy Clement
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/DeclareParentsMixin.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2006-2008 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/BindingScope.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AnnotationAJ.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2008 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
```

* Contributors:
*   Andy Clement   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/WeakClassLoaderReference.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelWeakClassLoaderReference.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/WildcardedUnresolvedType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ArrayReferenceType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelWeavingSupport.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/AnnotationAccessFieldVar.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/IWeavingSupport.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ExactAnnotationFieldTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/BindingAnnotationFieldTypePattern.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005, 2017 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/Java15AnnotationFinder.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002, 2010 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:

* PARC initial implementation
* Nieraj Singh
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/BindingTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/TypePattern.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2003,2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Mik Kersten initial implementation
* Andy Clement Extensions for better IDE representation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/internal/Relationship.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Alexandre Vasseur initial implementation
* Adrian Colyer refactoring for traversal and grouping by kind
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PatternNodeVisitor.java
No license file was found, but licenses were detected in source scan.

/***

* Copyright (c) 2012 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Abraham Nevado (lucierna) initial implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/SimpleCache.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/SimpleCacheFactory.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005,2018 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/ClassPreProcessorAgentAdapter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/ClassPreProcessor.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2002-2009 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   PARC   initial implementation
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/CrosscuttingMembersSet.java
No license file was found, but licenses were detected in source scan.

```
/* ====================================================================
* The Apache Software License, Version 1.1
*
* Copyright (c) 2001 The Apache Software Foundation.  All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*       "This product includes software developed by the
*        Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation" and
*    "Apache BCEL" must not be used to endorse or promote products
*    derived from this software without prior written permission. For
*    written permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache",
*    "Apache BCEL", nor may "Apache" appear in their name, without
*    prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED.  IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
```

* SUCH DAMAGE.
* ==
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.
*/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/BootstrapMethods.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Deprecated.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ExceptionTable.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/Repository.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/BranchHandle.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/LineNumberGen.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/FieldGen.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/BasicType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantInteger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/NestHost.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantInvokeDynamic.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionLV.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/SwitchBuilder.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/IINC.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/util/NonCachingClassLoaderRepository.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantCP.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/StackMapEntry.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantLong.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantString.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/LOOKUPSWITCH.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionShort.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/ClassGen.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/util/SyntheticRepository.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/MethodGen.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantFieldref.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantObject.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantMethodHandle.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/util/ClassLoaderReference.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/SimpleConstant.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/FieldOrMethod.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Utility.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Synthetic.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/MULTIANEWARRAY.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/JavaClass.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/StackMapType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/Constants.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/ConstantsInitializer.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/CodeExceptionGen.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InvokeDynamic.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/FieldOrMethod.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/util/Repository.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantFloat.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/CodeException.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/Type.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/ExceptionConstants.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/TargetLostException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/SourceFile.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantNameAndType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Attribute.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantDouble.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Code.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionHandle.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/Instruction.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/util/ClassLoaderRepository.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/ReturnaddressType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Field.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/RET.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/LineNumber.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantMethodref.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/util/ThreadLocalAwareRepository.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantMethodType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionTargeter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/LocalVariableGen.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/ObjectType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/ClassGenException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/InnerClass.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/FieldGenOrMethodGen.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Unknown.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/TABLESWITCH.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantPool.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/NestMembers.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InvokeInstruction.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ClassFormatException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantDynamic.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Node.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/LocalVariable.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/INVOKEINTERFACE.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/util/ByteSequence.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ClassVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/LineNumberTable.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/ArrayType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionByte.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionSelect.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Constant.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionList.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantInterfaceMethodref.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/InnerClasses.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/ReferenceType.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionConstants.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Modifiers.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/FieldInstruction.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/StackMap.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/util/DefaultClassLoaderReference.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantClass.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantUtf8.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionCP.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/LocalVariableTable.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ClassParser.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/InstructionBranch.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/Method.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   PARC   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelClassWeaver.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AndTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ExposedState.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/DeclareSoft.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/IClassWeaver.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/model/AsmRelationshipProvider.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/BasicTokenSource.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/Advice.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/NewMethodTypeMunger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/CrosscuttingMembers.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ResolvedPointcutDefinition.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/WildTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ModifiersPattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/IHasSourceLocation.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/Shadow.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/Bindings.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/IToken.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/Position.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/WithinPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ParserException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/MemberKind.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/Range.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelVar.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/NameMangler.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/Var.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/And.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ReferencePointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/BasicToken.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/ast/Not.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/NameBindingPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/SignaturePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ShadowMunger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/SourceContextImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/MemberImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AjAttribute.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/IHasPosition.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/Declare.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ITokenSource.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/IntMap.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/PrivilegedAccessMunger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/Pointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/Expr.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/OrTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AndPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/TypePatternQuestions.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelMethod.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelCflowStackFieldAdder.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/KindedPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ISourceContext.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/NewFieldTypeMunger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ExactTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/NotPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/ast/Literal.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/Lint.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AjcMemberMaker.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/BCException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelRenderer.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ConcreteCflowPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/OrPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/DeclareErrorOrWarning.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/NewConstructorTypeMunger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelFieldRef.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/FormalBinding.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/Iterators.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PerFromSuper.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ThisOrTargetPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ThrowsPattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/HandlerPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/UnwovenClassFile.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PatternNode.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/Test.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/Instanceof.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PerSingleton.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/WithincodePointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelField.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/ExceptionRange.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/ast/CallExpr.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/DeclarePrecedence.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/TypePatternList.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/Call.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/SimpleScope.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/FieldGetCall.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/IExprVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelCflowAccessVar.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/NotTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/LazyMethodGen.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/CflowPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/Checker.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/Or.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/StaticJoinPointFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/Utility.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PerCflow.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ConcreteTypeMunger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/IScope.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/NewParentTypeMunger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PerClause.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/FieldGet.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/NamePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/ASTNode.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AdviceKind.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/bcel/ShadowRange.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ArgsPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PerObject.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ast/ITestVisitor.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2005 IBM Corporation
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Adrian Colyer     initial implementation
*      Andy Clement     got it working
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/DeclareAnnotation.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2004 IBM Corporation
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Matthew Webster, Adrian Colyer,
*     Martin Lippert     initial implementation
*     Andy Clement
*     Roy Varghese - Bug 473555
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/ExtensibleURLClassLoader.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
```

* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* initial implementation Alexandre Vasseur
* generic signature update Adrian Colyer
**********/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/Aspects.java
No license file was found, but licenses were detected in source scan.

/* **********
* Copyright (c) 2004,2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Matthew Webster, IBM
* **********/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/Dump.java
No license file was found, but licenses were detected in source scan.

/* **********
* Copyright (c) 2010 SpringSource, Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **********/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/NewMemberClassTypeMunger.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2008 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/GeneratedReferenceTypeDelegate.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/Utils.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2006 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Adrian Colyer   Initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/InternalUseOnlyPointcutParser.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/DeferredResolvedPointcutDefinition.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement   Initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/TypeVariableDeclaringElement.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 1999-2001 Xerox Corporation,
*               2002 Palo Alto Research Center, Incorporated (PARC),
*               2006 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Xerox/PARC     initial implementation
* ******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/InitializerSignature.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* ******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/JavaLangTypeToResolvedTypeConverter.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2004 IBM Corporation
*
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
```

* Andy Clement initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadStackFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/AnnotationDefault.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadStackFactoryImpl11.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/EnclosingMethod.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadStack.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadCounter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/CFlowCounter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadStackFactoryImpl.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2009 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Andy Clement initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ResolvableTypeList.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:

```
*     PARC     initial implementation
*     Alexandre Vasseur    support for @AJ aspects
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelAdvice.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelShadow.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2004 IBM Corporation
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*    Andy Clement IBM     initial implementation 30-May-2004
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/WeaveMessage.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 1999-2001 Xerox Corporation,
*               2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/GenericSignature.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 1998-2001 Xerox Corporation,
*               2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
```

* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Xerox/PARC   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/Version.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2017 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/SoftHashMap.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Alexandre Vasseur   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/MethodDelegateTypeMunger.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2010 Contributors
* All rights reserved.

* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Andy Clement (SpringSource)
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ConstantPoolReader.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/CompressingDataOutputStream.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelConstantPoolWriter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ConstantPoolWriter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelConstantPoolReader.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement, SpringSource
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/TypeDelegateResolver.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:

* initial implementation Andy Clement
**********/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/FakeAnnotation.java
No license file was found, but licenses were detected in source scan.

/* ==
* The Apache Software License, Version 1.1
*
* Copyright (c) 2001, 2017 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation" and
* "Apache BCEL" must not be used to endorse or promote products
* derived from this software without prior written permission. For
* written permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache",
* "Apache BCEL", nor may "Apache" appear in their name, without
* prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT

* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ==
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.
*/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/util/ClassPath.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2006 IBM Corporation and others.
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement - initial implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/LockSignature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/UnlockSignatureImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/UnlockSignature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/LockSignatureImpl.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt

*
* Contributors:
* initial implementation Adrian Colyer
**/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclareParents.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2003 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Mik Kersten initial implementation
* Andy Clement incremental support and switch on/off state
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/AsmManager.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2003 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Mik Kersten initial implementation
* Andy Clement Extensions for better IDE representation
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/internal/AspectJElementHierarchy.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2002 Palo Alto Research Center, Incorporated (PARC).

* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   PARC   initial implementation
*   Alexandre Vasseur   rearchitected for #75442 finer grained matching
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/PerObjectInterfaceTypeMunger.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/DeclareTypeErrorOrWarning.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005 IBM
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PerTypeWithin.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AnnotationOnTypeMunger.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2011 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement   Initial implementation
* ******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/HasMemberTypePatternForPerThisMatching.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2002, 2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     PARC     initial implementation
* ******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/NoTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/EllipsisTypePattern.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2007 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
```

* Contributors:
*   Linton Ye https://bugs.eclipse.org/bugs/show_bug.cgi?id=193065
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/CustomMungerFactory.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2003 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Xerox/PARC   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/IProgressListener.java
No license file was found, but licenses were detected in source scan.

/* ==
* The Apache Software License, Version 1.1
*
* Copyright (c) 2016-17 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in
*   the documentation and/or other materials provided with the
*   distribution.
*
* 3. The end-user documentation included with the redistribution,
*   if any, must include the following acknowledgment:
*   "This product includes software developed by the
*   Apache Software Foundation (http://www.apache.org/)."

```
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation" and
*    "Apache BCEL" must not be used to endorse or promote products
*    derived from this software without prior written permission. For
*    written permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache",
*    "Apache BCEL", nor may "Apache" appear in their name, without
*    prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED.  IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation.  For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.
*/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-
jar/org/aspectj/apache/bcel/classfile/Module.java
No license file was found, but licenses were detected in source scan.

/* ********************************************************************
* Copyright (c) 1999-2001 Xerox Corporation,
*               2002-2018 Palo Alto Research Center, Incorporated (PARC), Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Xerox/PARC    initial implementation
```

*   Alex Vasseur   new factory methods for variants of JP
* Abraham Nevado   new factory methods for collapsed SJPs
*   Andy Clement   new factory methods that rely on LDC <class>
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/Factory.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 1999-2001 Xerox Corporation,
*   2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Xerox/PARC   initial implementation
*   Alex Vasseur   wired up for @AJ proceeding
*   Andy Clement 23-06-06 added extras for @AJ
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/AroundClosure.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2019 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/WildChildFinder.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2008 Contributors
*

* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement - refactored out of MemberImpl
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/SignatureUtils.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2006 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/IfFinder.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2018 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/RuntimeVersion.java
No license file was found, but licenses were detected in source scan.

/**
* Copyright (c) 2005 Contributors.

* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   David Knibb   initial implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/IWeavingContext.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/DefaultWeavingContext.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2008 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/IStructureModel.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2004 IBM Corporation
*
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement   initial implementation
*   Heavily based on LocalVariableTable
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/LocalVariableTypeTable.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
* Copyright (c) 2004 IBM Corporation and others.
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     IBM Corporation - initial API and implementation
*      (Andy Clement)
*******************************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelCflowCounterFieldAdder.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2014 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/RequiredTypes.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2004 IBM Corporation.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/patterns/WithinCodeAnnotationPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/StandardPointcutExpression.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/BindingAnnotationTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/NotAnnotationTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AnnotatedElement.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AndAnnotationTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ArgsAnnotationPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/OrAnnotationTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ThisOrTargetAnnotationPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ExactAnnotationTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/BindingPattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/PointcutExpression.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PointcutEvaluationExpenseComparator.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/internal/tools/TypePatternMatcherImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/WithinAnnotationPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/internal/tools/StandardPointcutExpressionImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AnnotationPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AnnotationPatternList.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/TypePatternMatcher.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/PointcutPrimitive.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AnnotationTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/WildAnnotationTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/internal/tools/PointcutExpressionImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PointcutRewriter.java

No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002, 2010 Palo Alto Research Center, Incorporated (PARC) and others.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   PARC   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AnyTypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AnyWithAnnotationTypePattern.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2008 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Andy Clement
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/asm/AsmDetector.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement - SpringSource

* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/OrSignaturePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/NotSignaturePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AbstractSignaturePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AndSignaturePattern.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Alexandre Vasseur   initial implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/Aj.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/definition/Definition.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelAccessForInlineMunger.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/Options.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/Agent.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/ConcreteAspectCodeGen.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/DefaultMessageHandler.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/PerThisOrTargetPointcutVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclareAnnotation.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002 Contributors

* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   PARC   initial implementation
*   Andy Clement   pushed down into bcel module
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/Tag.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/generic/LocalVariableTag.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005, 2017 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/ClassLoaderWeavingAdaptor.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005-2008 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/AnnotationAccessVar.java

No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005-2012 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/TypeVariableReferenceType.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Ron Bodkin   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/IReflectionWorld.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2013 VMware
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement -   initial implementation {date}
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/RuntimeInvisTypeAnnos.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/RuntimeVisTypeAnnos.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/RuntimeTypeAnnos.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     PARC     initial implementation
*     Alexandre Vasseur    perClause support for @AJ aspects
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelWorld.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
* Copyright (c) 2009 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* initial implementation              Andy Clement
*******************************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclareMixin.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
* Copyright (c) 2012 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
```

*
* Contributors:
*   John Kew (vmware)   initial implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/CacheKeyResolver.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/CacheFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/DefaultCacheKeyResolver.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/DefaultCacheFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/CacheStatistics.java
No license file was found, but licenses were detected in source scan.

<!--
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*   contributors may be used to endorse or promote products derived from
*   this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
-->

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/signature/package.html
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/package.html
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2002-2019 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     PARC     initial implementation
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/WeaverStateInfo.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2004 IBM Corporation
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Matthew Webster, Adrian Colyer,
*     Martin Lippert     initial implementation
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/GeneratedClassHandler.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/WeavingClassLoader.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2005-2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
```

* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Andy Clement (IBM, SpringSource)
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/VersionedDataInputStream.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2012 VMware, Inc. custard
*
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Lyor Goldstein
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/FlatFileCacheBacking.java
No license file was found, but licenses were detected in source scan.

/* ==
* The Apache Software License, Version 1.1
*
* Copyright (c) 2017 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in
*   the documentation and/or other materials provided with the
*   distribution.

*
* 3. The end-user documentation included with the redistribution,
*   if any, must include the following acknowledgment:
*   "This product includes software developed by the
*   Apache Software Foundation (http://www.apache.org/)."
*   Alternately, this acknowledgment may appear in the software itself,
*   if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation" and
*   "Apache BCEL" must not be used to endorse or promote products
*   derived from this software without prior written permission. For
*   written permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache",
*   "Apache BCEL", nor may "Apache" appear in their name, without
*   prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ==
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.
*/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantPackage.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ModuleMainClass.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ModulePackages.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/ConstantModule.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Andy Clement, SpringSource
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/MemberUtils.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 1999-2001 Xerox Corporation,
*               2002 Palo Alto Research Center, Incorporated (PARC).
*               2018 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Xerox/PARC     initial implementation
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/LangUtil.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2013 VMware
*
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*    Andy Clement     initial implementation
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/TypeAnnotationGen.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/MethodParameters.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2003,2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Mik Kersten     initial implementation
*     Andy Clement
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/internal/RelationshipMap.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/IHierarchy.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/HierarchyWalker.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 1999-2001 Xerox Corporation,
*               2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Xerox/PARC     initial implementation
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/ICommand.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/IMessageHolder.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/FieldSignature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/CatchClauseSignatureImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/PartialOrder.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/FieldSignatureImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/FuzzyBoolean.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/SourceLocation.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/CodeSignatureImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/StringMaker.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/JoinPointImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/ISourceLocation.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/InitializerSignatureImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/CFlow.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/IMessage.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/ConstructorSignature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/MessageHandler.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/JoinPoint.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/Signature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/ReflectionFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/MemberSignature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/AdviceSignatureImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/SourceLocation.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/NoAspectBoundException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/SourceLocationImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/CountingMessageHandler.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/MessageUtil.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/CatchClauseSignature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/PerObjectMap.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/MemberSignatureImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/AbortException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/SignatureImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/MessageWriter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/CodeSignature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/ConstructorSignatureImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/IMessageHandler.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/CFlowStack.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/MethodSignature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/Message.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/AdviceSignature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/CFlowPlusState.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/reflect/MethodSignatureImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/Conversions.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/FileUtil.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2006 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Adrian Colyer Initial implementation

* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/ReflectionFastMatchInfo.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/ILifecycleAware.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2004, 2017 Contributors
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   IBM Corporation - initial API and implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/PointcutParser.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2003 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Isberg   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/UtilClassLoader.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2006 Contributors. All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt

*
* Contributors: IBM Corporation - initial API and implementation
*   Helen Hawkins   - iniital version
**/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/IEclipseSourceContext.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002 Palo Alto Research Center, Incorporated (PARC).
*   2004 IBM Corporation.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/FuzzyBoolean.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   PARC   initial implementation
*   Andy Clement - June 2005 - separated out from ResolvedType
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AbstractReferenceTypeDelegate.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ReferenceTypeDelegate.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ReferenceType.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2004, 2013 IBM Corporation
*
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*    Andy Clement     initial implementation
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/AnnotationGen.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2002-2019 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/DeclareParents.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2004 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Jim Hugunin     initial implementation
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/FastMatchInfo.java

No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2003 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Mik Kersten   initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/IRelationship.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/IProgramElement.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/IElementHandleProvider.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/IHierarchyListener.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/IRelationshipMap.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors. All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Helen Hawkins - Initial implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AnnotationTargetKind.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2006 IBM
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at

* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement - initial API and implementation
**/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/PoliceExtensionUse.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2005 IBM
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement   initial implementation
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/TypeAnnotationAccessVar.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2002 Palo Alto Research Center, Incorporated (PARC).
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   PARC   initial implementation
*   Alexandre Vasseur   if() implementation for @AJ style
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/IfPointcut.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2010 Contributors

* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement
*   Nieraj Singh
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/TypeCategoryTypePattern.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Alexandre Vasseur   initial implementation
*   Abraham Nevado - Lucierna simple caching strategy
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/definition/DocumentParser.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Ron Bodkin Initial implementation
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/ltw/LTWWorld.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2004 IBM Corporation
*
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*    Andy Clement     initial implementation
*         Copied from bits of original CFlowStack
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadCounterImpl11.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/runtime/internal/cflowstack/ThreadStackImpl11.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2006 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Andy Clement IBM     initial implementation
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/EnumAnnotationValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/IModelFilter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ArrayAnnotationValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AnnotationAnnotationValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/IVerificationRequired.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/SimpleAnnotationValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AnnotationNameValuePair.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/AnnotationValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ClassAnnotationValue.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2005 IBM, Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Andy Clement     initial implementation
* ******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/PerTypeWithinTargetTypeMunger.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************
* Copyright (c) 2006 Contributors. All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors: IBM Corporation - initial API and implementation
*      Helen Hawkins   - initial version
*******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/internal/HandleProviderDelimiter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/internal/NameConvertor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/internal/CharOperation.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/asm/internal/JDTLikeHandleProvider.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/model/AsmRelationshipUtils.java

No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005-2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/UnresolvedTypeVariableReferenceType.java
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//   notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//   notice, this list of conditions and the following disclaimer in the
//   documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//   this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/AnnotationVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/Frame.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/TypePath.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/Handle.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/ClassReader.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/signature/SignatureReader.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/FieldVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/Type.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/CurrentFrame.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/aj/org/objectweb/asm/ModuleVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/ByteVector.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/MethodTooLargeException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/MethodWriter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/ModuleWriter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/TypeReference.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/ClassWriter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/FieldWriter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/Context.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/signature/SignatureVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/RecordComponentVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/ConstantDynamic.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/ClassVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/Attribute.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/AnnotationWriter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/Handler.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/MethodVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/Symbol.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/RecordComponentWriter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/SymbolTable.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/signature/SignatureWriter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/ClassTooLargeException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/Edge.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/Opcodes.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/aj/org/objectweb/asm/Label.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/aj/org/objectweb/asm/Constants.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
* Copyright (c) 2012 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   John Kew (vmware)         initial implementation
*   Lyor Goldstein (vmware) add support for weaved class being re-defined
*******************************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/CachedClassEntry.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2005-2017 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/ArgNameFinder.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/ReflectionWorld.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2005-2008 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/util/GenericSignatureParser.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
* Copyright (c) 2004-2019 Contributors
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*******************************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/WeaverMessages.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2002-2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     PARC               initial implementation
*     Andy Clement  6Jul05 generics - signature attribute
*     Abraham Nevado
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/LazyClassGen.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
* Copyright (c) 2012 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   John Kew (vmware)         initial implementation
```

*   Lyor Goldstein (vmware) add support for weaved class being re-defined
**********/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/CachedClassReference.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/CacheBacking.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/WeavedClassCache.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/AbstractCacheBacking.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/GeneratedCachedClassHandler.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/DefaultFileCacheBacking.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/AbstractFileCacheBacking.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/cache/AbstractIndexedFileCacheBacking.java
No license file was found, but licenses were detected in source scan.

/* **********
* Copyright (c) 2010 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **********/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/BoundedReferenceType.java
No license file was found, but licenses were detected in source scan.

/* **********
* Copyright (c) 2005, 2017 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* **********/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/weaver/reflect/ReflectionBasedResolvedMemberImpl.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2009 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/PersistenceSupport.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2004 IBM Corporation
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Matthew Webster, Adrian Colyer, John Kew + Lyor Goldstein (caching)
*   Martin Lippert   initial implementation
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/WeavingAdaptor.java
No license file was found, but licenses were detected in source scan.

/* ***
* Copyright (c) 2002, 2017 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
* Palo Alto Research Center, Incorporated (PARC).
* ***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/ClassPathManager.java
No license file was found, but licenses were detected in source scan.

```
/*******************************************************************************
* Copyright (c) 2004 IBM
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*    Andy Clement - initial API and implementation
*******************************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/Constants.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2002-2010
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* ********************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/Member.java
No license file was found, but licenses were detected in source scan.

```
/* ********************************************************************
* Copyright (c) 2004, 2013 IBM, VMware
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     Andy Clement -     initial implementation {date}
* ********************************************************************/
```

Found in path(s):

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/ElementValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/RuntimeAnnos.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/RuntimeInvisAnnos.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/RuntimeInvisParamAnnos.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005-2012 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Adrian Colyer   Initial implementation
*   Andy Clement   various fixes
*   Trask Stanalker #373195
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/context/CompilationAndWeavingContext.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2000, 2003 IBM Corporation and others.
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   IBM Corporation - initial API and implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ICrossReferenceHandler.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2004 IBM
* All rights reserved.

* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Andy Clement -   initial implementation {date}
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/AnnotationElementValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/EnumElementValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/RuntimeVisParamAnnos.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/SimpleElementValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/RuntimeVisAnnos.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/RuntimeParamAnnos.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/ArrayElementValue.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/apache/bcel/classfile/annotation/ClassElementValue.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2011 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Abraham Nevado - Lucierna initial implementation
*   Just a slight variation of current DocumentParser.java from Alexandre Vasseur.
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/definition/SimpleAOPParser.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002,2005 Contributors

* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   PARC   initial implementation
*   Andy Clement start of generics upgrade...
*   Adrian Colyer - overhaul
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/UnresolvedType.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2005, 2017 Contributors.
*
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/AnnotationFinder.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2004 IBM Corporation
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Matthew Webster, Adrian Colyer,
*   Martin Lippert   initial implementation
*   Andy Clement
*   Abraham Nevado
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/loadtime/WeavingURLClassLoader.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2002-2006 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*     PARC     initial implementation
*     AndyClement extracted as self contained type from Lint type (4-Aug-06)
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/LintMessage.java
No license file was found, but licenses were detected in source scan.

```
/* *******************************************************************
* Copyright (c) 2005 Contributors.
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Adrian Colyer   Initial implementation
* *******************************************************************/
```

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclareError.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/JoinPointSignatureIterator.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/ReflectionShadow.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/AdviceImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/InterTypeFieldDeclarationImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/ReflectionBasedReferenceTypeDelegate.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/DeclareAnnotation.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/NoSuchAdviceException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/StringToType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/DeclareErrorOrWarning.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/TypeVariablePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/MissingResolvedTypeWithKnownSignature.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/PerClauseImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/DeclarePrecedence.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/PerClause.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/AjTypeImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/PointcutImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/DeclareParents.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/JoinPointMatchImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcDeclareAnnotation.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/PointcutParameterImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/DeclarePrecedenceImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/StandardShadowMatchImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/Advice.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/TypePatternBasedPerClause.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/HasMemberTypePatternFinder.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/InterTypeConstructorDeclaration.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/ContextBasedMatcher.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/SignaturePatternImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/Java14GenericSignatureInformationProvider.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/StandardShadow.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/Java15ReflectionBasedReferenceTypeDelegate.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/GenericSignatureInformationProvider.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/TypePatternImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/PointcutBasedPerClauseImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/Pointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/SignaturePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/Java15GenericSignatureInformationProvider.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/context/ContextToken.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/DeclareErrorOrWarningImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/InterTypeMethodDeclaration.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/ReflectionBasedReferenceTypeDelegateFactory.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcDeclareEoW.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/ShadowMatch.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/ScopeWithTypeVariables.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcPrivileged.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/DeclareAnnotationImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/PointcutExpression.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/PointcutParameter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/DeclareParentsImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/InterTypeConstructorDeclarationImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/JoinPointMatch.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/UnsupportedPointcutPrimitiveException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/JoinPointSignature.java

* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/NoSuchPointcutException.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/AjTypeSystem.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/internal/tools/MatchingContextBasedTest.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/InterTypeDeclarationImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/MatchingContext.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/HasThisTypePatternTriedToSneakInSomeGenericOrParameterizedTypePatternMatchingStuffAnywhereVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcITD.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/TypeVariableReference.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/AbstractPatternNodeVisitor.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/PerClauseKind.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelGenericSignatureToTypeXConverter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/AjType.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/TypePattern.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/InterTypeDeclaration.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/ReflectionVar.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/PointcutExpressionImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/context/ContextFormatter.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcDeclarePrecedence.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/bridge/context/PinpointingMessageHandler.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/TypePatternBasedPerClauseImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/internal/tools/PointcutDesignatorHandlerBasedPointcut.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/DefaultMatchingContext.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/InterTypeFieldDeclaration.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-

jar/org/aspectj/lang/reflect/PointcutBasedPerClause.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/reflect/ShadowMatchImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/DeclareSoftImpl.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcDeclareParents.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/DeclareWarning.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/annotation/ajcDeclareSoft.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/tools/PointcutDesignatorHandler.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/AdviceKind.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/reflect/DeclareSoft.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/patterns/TypeVariablePatternList.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/internal/lang/reflect/InterTypeMethodDeclarationImpl.java
No license file was found, but licenses were detected in source scan.

/***
* Copyright (c) 2005 Contributors.
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution, and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:
*   Adrian Colyer   initial implementation
***/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/lang/annotation/AdviceName.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
*
* Contributors:

* PARC initial implementation
* RonBodkin/AndyClement optimizations for memory consumption/speed
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelObjectType.java
No license file was found, but licenses were detected in source scan.

/* **
* Copyright (c) 2002-2010 Contributors
* All rights reserved.
* This program and the accompanying materials are made available
* under the terms of the Eclipse Public License v 2.0
* which accompanies this distribution and is available at
* https://www.eclipse.org/org/documents/epl-2.0/EPL-2.0.txt
* **/

Found in path(s):
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/bcel/BcelWeaver.java
* /opt/cola/permits/1337070282_1654119253.2769773/0/aspectjweaver-1-9-7-sources-jar/org/aspectj/weaver/ResolvedMemberImpl.java

1.157 json-smart 2.4.8

1.157.1 Available under license :

No license file was found, but licenses were detected in source scan.

<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>

Found in path(s):
* /opt/cola/permits/1340076263_1654622360.233129/0/json-smart-2-4-8-2-jar/META-INF/maven/net.minidev/json-smart/pom.xml
No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bnd-LastModified: 1644737862078
Build-Jdk: 11.0.14
Built-By: uriel
Bundle-Description: JSON (JavaScript Object Notation) is a lightweight
 data-interchange format. It is easy for humans to read and write. It
 is easy for machines to parse and generate. It is based on a subset
of the JavaScript Programming Language, Standard ECMA-262 3rd Edition
 - December 1999. JSON is a text format that is completely language i
ndependent but uses conventions that are familiar to programmers of t
he C-family of languages, including C, C++, C#, Java, JavaScript, Per

l, Python, and many others. These properties make JSON an ideal data-
interchange language.
Bundle-DocURL: https://urielch.github.io/
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.txt
Bundle-ManifestVersion: 2
Bundle-Name: json-smart
Bundle-SymbolicName: net.minidev.json-smart
Bundle-Vendor: Chemouni Uriel
Bundle-Version: 2.4.8
Created-By: Apache Maven Bundle Plugin
Export-Package: net.minidev.json;uses:="net.minidev.asm,net.minidev.js
on.parser,net.minidev.json.reader,net.minidev.json.writer";version="2
.4.8",net.minidev.json.annotate;version="2.4.8",net.minidev.json.pars
er;uses:="net.minidev.json.writer";version="2.4.8",net.minidev.json.r
eader;uses:="net.minidev.json";version="2.4.8",net.minidev.json.write
r;uses:="net.minidev.json,net.minidev.json.parser";version="2.4.8"
Import-Package: net.minidev.asm;version="[2.4,3)",net.minidev.json,net
.minidev.json.annotate,net.minidev.json.parser,net.minidev.json.reade
r,net.minidev.json.writer
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.8))"
Tool: Bnd-5.1.1.202006162103

Found in path(s):
* /opt/cola/permits/1340076263_1654622360.233129/0/json-smart-2-4-8-2-jar/META-INF/MANIFEST.MF

1.158 brave-instrumentation-rpc-adapters 5.13.9

1.158.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all

other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Brave

Copyright 2013-2020 The OpenZipkin Authors

This product contains a modified portion of 'InetAddresses', distributed
by Google in the Guava Library:

* Copyright (C) 2008 The Guava Authors
* License: Apache License v2.0
* Homepage: https://github.com/google/guava

This product contains a modified portion of 'JsonUtf8Writer', distributed
by Square, Inc in the Moshi Library:

* Copyright (C) 2010 Google Inc.
* License: Apache License v2.0
* Homepage: https://github.com/square/moshi

This product contains a modified portion of 'WeakConcurrentMap', distributed
by Rafael Winterhalter in the weak-lock-free Library:

* License: Apache License v2.0
* Homepage: https://github.com/raphw/weak-lock-free

1.159 spring-boot-webflux-starter 2.7.2

1.159.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only

on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Spring Boot 2.7.2
Copyright (c) 2012-2022 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

1.160 junit-jupiter-junit-jupiter-engine 5.8.2

1.160.1 Available under license :

Eclipse Public License - v 2.0
==========

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM

CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:
* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **b)** in the case of each subsequent Contributor:
 * **i)** changes to the Program, and
 * **ii)** additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

3.1 If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
 * **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 * **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 * **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
 * **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available

under the terms of such Secondary Licenses, and
* **b)** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR

OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
Open Source Licenses
==========

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.161 liquibase 4.9.1

1.161.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that

remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.162 mac-os 4.1.79.Final

1.162.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2021 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,

~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~   https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->

Found in path(s):
* /opt/cola/permits/1519136955_1671710602.1421163/0/netty-resolver-dns-classes-macos-4-1-79-final-sources-2-jar/META-INF/maven/io.netty/netty-resolver-dns-classes-macos/pom.xml
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136955_1671710602.1421163/0/netty-resolver-dns-classes-macos-4-1-79-final-sources-2-jar/io/netty/resolver/dns/macos/package-info.java
* /opt/cola/permits/1519136955_1671710602.1421163/0/netty-resolver-dns-classes-macos-4-1-79-final-sources-2-jar/io/netty/resolver/dns/macos/DnsResolver.java
* /opt/cola/permits/1519136955_1671710602.1421163/0/netty-resolver-dns-classes-macos-4-1-79-final-sources-2-jar/io/netty/resolver/dns/macos/MacOSDnsServerAddressStreamProvider.java

1.163 mockito-junit-jupiter 4.5.1

1.163.1 Available under license :

The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.164 jetbrains-annotations 17.0.0

1.164.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2000-2016 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/jetbrains/annotations/Contract.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2000-2014 JetBrains s.r.o.

*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/intellij/lang/annotations/MagicConstant.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/jetbrains/annotations/Nullable.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2000-2015 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/jetbrains/annotations/TestOnly.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/intellij/lang/annotations/Flow.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/jetbrains/annotations/Nls.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006 Sascha Weinreuter

```
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/intellij/lang/annotations/Identifier.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/intellij/lang/annotations/RegExp.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/intellij/lang/annotations/Subst.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/intellij/lang/annotations/Language.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/intellij/lang/annotations/Pattern.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/intellij/lang/annotations/PrintFormat.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2000-2012 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/jetbrains/annotations/NotNull.java

* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/intellij/lang/annotations/JdkConstants.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2000-2009 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/jetbrains/annotations/PropertyKey.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-zip/org/jetbrains/annotations/NonNls.java

1.165 asm-commons 9.2

1.165.1 Available under license :

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*   contributors may be used to endorse or promote products derived from
*   this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"

* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/package.html
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//   notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//   notice, this list of conditions and the following disclaimer in the
//   documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//   this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/TryCatchBlockSorter.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/AnalyzerAdapter.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/SignatureRemapper.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/MethodRemapper.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/ModuleHashesAttribute.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/StaticInitMerger.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/AdviceAdapter.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/ClassRemapper.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/ModuleRemapper.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-

jar/org/objectweb/asm/commons/RecordComponentRemapper.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/ModuleResolutionAttribute.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/Method.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/InstructionAdapter.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/JSRInlinerAdapter.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/FieldRemapper.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/SimpleRemapper.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/ModuleTargetAttribute.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/TableSwitchGenerator.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/Remapper.java
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/CodeSizeEvaluator.java
No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//    notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//    notice, this list of conditions and the following disclaimer in the
//    documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//    this software without specific prior written permission.
/**
  * Constructs a new {@link SerialVersionUIDAdder}. <i>Subclasses must not use this
  * constructor</i>. Instead, they must use the {@link #SerialVersionUIDAdder(int, ClassVisitor)}
  * version.
  *
  * @param classVisitor a {@link ClassVisitor} to which this visitor will delegate calls.
  * @throws IllegalStateException If a subclass calls this constructor.
  */
```

Found in path(s):
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/SerialVersionUIDAdder.java
No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//    notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//    notice, this list of conditions and the following disclaimer in the
//    documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//    this software without specific prior written permission.
/**
  * Constructs a new {@link GeneratorAdapter}. <i>Subclasses must not use this constructor</i>.
  * Instead, they must use the {@link #GeneratorAdapter(int, MethodVisitor, int, String, String)}
  * version.
  *
  * @param methodVisitor the method visitor to which this adapter delegates calls.
  * @param access the method's access flags (see {@link Opcodes}).
  * @param name the method's name.
  * @param descriptor the method's descriptor (see {@link Type}).
  * @throws IllegalStateException if a subclass calls this constructor.
  */
```

Found in path(s):
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/GeneratorAdapter.java
No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//    notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//    notice, this list of conditions and the following disclaimer in the
//    documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//    this software without specific prior written permission.
/**
  * Constructs a new {@link LocalVariablesSorter}. <i>Subclasses must not use this constructor</i>.
  * Instead, they must use the {@link #LocalVariablesSorter(int, int, String, MethodVisitor)}
  * version.
  *
  * @param access access flags of the adapted method.
  * @param descriptor the method's descriptor (see {@link Type}).
  * @param methodVisitor the method visitor to which this adapter delegates calls.
  * @throws IllegalStateException if a subclass calls this constructor.
```

```
 */
```

Found in path(s):
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/LocalVariablesSorter.java
No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//    notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//    notice, this list of conditions and the following disclaimer in the
//    documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//    this software without specific prior written permission.
/**
 * Constructs a new {@link AnnotationRemapper}. <i>Subclasses must not use this constructor</i>.
 * Instead, they must use the {@link #AnnotationRemapper(int,String,AnnotationVisitor,Remapper)}
 * version.
 *
 * @param descriptor the descriptor of the visited annotation. May be {@literal null}.
 * @param annotationVisitor the annotation visitor this remapper must delegate to.
 * @param remapper the remapper to use to remap the types in the visited annotation.
 */
```

Found in path(s):
* /opt/cola/permits/1288694265_1647366206.14/0/asm-commons-9-2-sources-jar/org/objectweb/asm/commons/AnnotationRemapper.java

1.166 fabric8-::-kubernetes-model-::-metrics 5.12.1

1.166.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*        http://www.apache.org/licenses/LICENSE-2.0
```

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340815683_1654860821.6094427/0/kubernetes-model-metrics-5-12-1-sources-jar/manifest.vm
No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

       http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815683_1654860821.6094427/0/kubernetes-model-metrics-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-metrics/pom.xml

1.167 fabric8-::-kubernetes-model-::-policy 5.12.1

1.167.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.

You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340816099_1654860760.566453/0/kubernetes-model-policy-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-policy/pom.xml
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340816099_1654860760.566453/0/kubernetes-model-policy-5-12-1-sources-jar/manifest.vm

1.168 apache-log4j-api 2.17.2

1.168.1 Available under license :

Apache Log4j Core
Copyright 1999-2012 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache Log4j
Copyright 1999-2021 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

TypeUtil.java
Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

picocli (http://picocli.info)
Copyright 2017 Remko Popma
Apache Log4j
Copyright 1999-2012 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

Maven Wrapper Jar
Copyright 2016-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent

to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,

excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any

risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.169 fabric8-::-kubernetes-model-::-discovery 5.12.1

1.169.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

Copyright (C) 2015 Red Hat, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815927_1654860863.8588238/0/kubernetes-model-discovery-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-discovery/pom.xml
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and

* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340815927_1654860863.8588238/0/kubernetes-model-discovery-5-12-1-sources-jar/manifest.vm

1.170 javabeans-activation-framework-api 1.2.0

1.170.1 Available under license :

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or

modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software

(or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under

the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new

license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly

infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be

unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such

interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in

accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute

so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free

programs whose distribution conditions are different, write to the
author to ask for permission. For software which is copyrighted by the
Free Software Foundation, write to the Free Software Foundation; we
sometimes make exceptions for this. Our decision will be guided by the
two goals of preserving the free status of all derivatives of our free
software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR
DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL
DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

> One line to give the program's name and a brief idea of what it does.
> Copyright (C) <year> <name of author>
>
> This program is free software; you can redistribute it and/or modify
> it under the terms of the GNU General Public License as published by
> the Free Software Foundation; either version 2 of the License, or

(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

#

Certain source files distributed by Oracle America, Inc. and/or its
affiliates are subject to the following clarification and special
exception to the GPLv2, based on the GNU Project exception for its

Classpath libraries, known as the GNU Classpath Exception, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

You should also note that Oracle includes multiple, independent programs in this software package. Some of those programs are provided under licenses deemed incompatible with the GPLv2 by the Free Software Foundation and others. For example, the package includes programs licensed under the Apache License, Version 2.0. Such programs are licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingle code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result. Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to further distribute the package.

Proceed with caution and we recommend that you obtain the advice of a lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party software.

CLASSPATH EXCEPTION
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or

based on this library. If you modify this library, you may extend this
exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement
from your version.

1.171 snappy-java 1.1.8.4

1.171.1 Available under license :

This product includes software developed by Google
Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
PureJavaCrc32C from apache-hadoop-common http://hadoop.apache.org/
(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by
"GCC RUntime Library Exception"
http://gcc.gnu.org/onlinedocs/libstdc++/manual/license.html

== Contributors ==
 * Tatu Saloranta
   * Providing benchmark suite
 * Alec Wysoker
   * Performance and memory usage improvement

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.172 okhttp 4.10.0

1.172.1 Available under license :

Note that publicsuffixes.gz is compiled from The Public Suffix List:
https://publicsuffix.org/list/public_suffix_list.dat

It is subject to the terms of the Mozilla Public License, v. 2.0:
https://mozilla.org/MPL/2.0/

1.173 junit 4.13.2

1.173.1 Available under license :

JUnit

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform,

distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the

Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement

and any licenses granted by Recipient relating to the Program shall continue
and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in
order to avoid inconsistency the Agreement is copyrighted and may only be
modified in the following manner. The Agreement Steward reserves the right to
publish new versions (including revisions) of this Agreement from time to time.
No one other than the Agreement Steward has the right to modify this Agreement.
The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to
serve as the Agreement Steward to a suitable separate entity. Each new version
of the Agreement will be given a distinguishing version number. The Program
(including Contributions) may always be distributed subject to the version of
the Agreement under which it was received. In addition, after a new version of
the Agreement is published, Contributor may elect to distribute the Program
(including its Contributions) under the new version. Except as expressly stated
in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to
the intellectual property of any Contributor under this Agreement, whether
expressly, by implication, estoppel or otherwise. All rights in the Program not
expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the
intellectual property laws of the United States of America. No party to this
Agreement will bring a legal action under this Agreement more than one year
after the cause of action arose. Each party waives its rights to a jury trial
in any resulting litigation.

1.174 reactive-streams 1.0.4

1.174.1 Available under license :

MIT-0

1.175 spring-cloud-sleuth-brave 3.1.3

1.175.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2013-2018 the original author or authors.
~
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~
~   https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software

~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.
-->

Found in path(s):
* /opt/cola/permits/1519136990_1671710579.6016715/0/spring-cloud-sleuth-brave-3-1-3-jar/META-INF/maven/org.springframework.cloud/spring-cloud-sleuth-brave/pom.xml

1.176 asm-analysis 9.1

1.176.1 Available under license :

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*   contributors may be used to endorse or promote products derived from
*   this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/package.html
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//   notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//   notice, this list of conditions and the following disclaimer in the
//   documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//   this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/BasicVerifier.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/Analyzer.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/SourceValue.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/SimpleVerifier.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/Interpreter.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/Value.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/AnalyzerException.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/Frame.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/BasicValue.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/Subroutine.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/SmallSet.java
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//   notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//   notice, this list of conditions and the following disclaimer in the
//   documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//   this software without specific prior written permission.

```
/**
  * Constructs a new {@link BasicInterpreter} for the latest ASM API version. <i>Subclasses must
  * not use this constructor</i>. Instead, they must use the {@link #BasicInterpreter(int)}
  * version.
  */
```

Found in path(s):
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/BasicInterpreter.java
No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//    notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//    notice, this list of conditions and the following disclaimer in the
//    documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//    this software without specific prior written permission.
/**
  * Constructs a new {@link SourceInterpreter} for the latest ASM API version. <i>Subclasses must
  * not use this constructor</i>. Instead, they must use the {@link #SourceInterpreter(int)}
  * version.
  */
```

Found in path(s):
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-jar/org/objectweb/asm/tree/analysis/SourceInterpreter.java

1.177 apache-commons-beanutils 1.9.4

1.177.1 Available under license :

Apache Commons BeanUtils
Copyright 2000-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent

to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,

excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any

risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.178 io-projectreactor 3.4.21

1.178.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2020-2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/Sinks.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SinksSpecs.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/ContextView.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2011-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and

* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFunctions.kt
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/package-info.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/package-info.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/package-info.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/function/package-info.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/package-info.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/package-info.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/concurrent/package-info.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/function/TupleExtensions.kt
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxExtensions.kt
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/adapter/package-info.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoExtensions.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2017-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-

jar/reactor/core/publisher/MonoPublishMulticast.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFilterWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/OperatorDisposables.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/concurrent/Queues.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/PeriodicWorkerTask.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/DelegateServiceScheduler.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/PeriodicSchedulerTask.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/SchedulerTask.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/WorkerTask.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxFilterWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxOnBackpressureBufferTimeout.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDelaySequence.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoExpand.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCacheTime.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/StrictSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxRefCountGrace.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/EmptyCompositeDisposable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxExpand.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxLimitRequest.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/Scannable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/Disposables.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDelayUntil.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/InstantPeriodicWorkerTask.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-

jar/reactor/core/CoreSubscriber.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2017-2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxWindowTimeout.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/Context5.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/Context4.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/Context.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2021-2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-

jar/reactor/core/publisher/FluxSwitchMapNoPrefetch.java
No license file was found, but licenses were detected in source scan.

Copyright (c) 2019-2021 VMware Inc. or its affiliates, All Rights Reserved.
#
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
#
https://www.apache.org/licenses/LICENSE-2.0
#
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
reactor.core.scheduler.ReactorBlockHoundIntegration

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/META-INF/services/reactor.blockhound.integration.BlockHoundIntegration
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/StateLogger.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2018-2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoUsingWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSwitchOnFirst.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxUsingWhen.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2015-2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/Context3.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/Context2.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSink.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/Context1.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/Context0.java

* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/ContextN.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (c) 2019-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelThen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ContextTrackingFunctionWrapper.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/context/CoreContext.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (c) 2019-2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/ReactorBlockHoundIntegration.java

* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/BoundedElasticScheduler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2016-2022 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxMap.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/Schedulers.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxMapSignal.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCreate.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/Operators.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoPeekTerminal.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/Flux.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxCreate.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDoOnEach.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCollectList.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSink.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SynchronousSink.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxHandleFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-

jar/reactor/core/publisher/FluxGenerate.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxProcessor.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SinkOneMulticast.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxReplay.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCollect.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/Mono.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDoFinally.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxBufferTimeout.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxMapFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxHandle.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/EmitterProcessor.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoIgnoreThen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDoFinally.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2016-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/DirectInnerContainer.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFromPublisher.java

* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/ExecutorScheduler.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoElementAt.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelPeek.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxElapsed.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoHandleFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoIgnorePublisher.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/InternalFluxOperator.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoRepeatWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoTakeLastOne.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoPeek.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSample.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/function/Tuple5.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/GroupedLift.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxNameFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/InternalMonoOperator.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxZip.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxRetry.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDetach.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDeferContextual.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxFlatMap.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxNever.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/function/Tuple3.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelMergeSort.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SignalPeek.java

* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSourceMonoFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSingle.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDematerialize.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ConnectableLift.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDelay.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/InnerProducer.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelMergeReduce.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxOnBackpressureDrop.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxRange.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxZipIterable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ImmutableSignal.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxRefCount.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxOnErrorResume.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxFlattenIterable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/function/Tuple7.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ConnectableLiftFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxFirstWithSignal.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSequenceEqual.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelCollect.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxRetryWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/LambdaSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelDoOnEach.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoLogFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoStreamCollector.java

* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSwitchIfEmpty.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFilter.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoTimeout.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDefer.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/GroupedLiftFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoAny.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxGroupJoin.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoContextWrite.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoProcessor.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoRetryWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSubscribeOnValue.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoMap.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelLog.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ReplayProcessor.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCallable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxWindowBoundary.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxWithLatestFrom.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxWindow.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDetach.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/Loggers.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxStream.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelFluxOnAssembly.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDoFirst.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDelaySubscription.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-

jar/reactor/core/publisher/MonoHide.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxAutoConnect.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxBufferWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxRepeatWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxEmpty.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxTakeUntil.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxHide.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/InternalConnectableFluxOperator.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFlattenIterable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/SingleWorkerScheduler.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/function/Tuple4.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxBufferBoundary.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/function/Tuple6.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelFilter.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SerializedSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SignalLogger.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoLog.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelFlux.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxMerge.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxPeek.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/UnicastProcessor.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/Exceptions.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxTakeUntilOther.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxInterval.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-

jar/reactor/core/publisher/FluxScanSeed.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoZip.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDoFirstFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxJoin.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxLiftFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSwitchIfEmpty.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxIndexFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoError.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoHandle.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ConnectableFlux.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoErrorSupplied.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSupplier.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/LambdaMonoSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDelayElement.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelFluxHide.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/Scheduler.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/BlockingOptionalMonoSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxCancelOn.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxTakeFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoRetry.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoOperator.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/concurrent/SpscArrayQueue.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SignalType.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/adapter/JdkFlowAdapter.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-

jar/reactor/core/publisher/MonoDoFirstFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxBuffer.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoBridges.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoName.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDoOnEachFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/ParallelScheduler.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSourceFlux.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/DirectProcessor.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFirstWithSignal.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelSource.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoNameFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoRepeat.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSourceFluxFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/BlockingFirstSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoReduceSeed.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxFromMonoOperator.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSwitchMap.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCallableOnAssembly.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDistinct.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/CorePublisher.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFlatMap.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxMaterialize.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoOnErrorResume.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxCallable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-

jar/reactor/core/publisher/FluxCallableOnAssembly.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSampleFirst.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSkipUntil.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxLift.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSkipWhile.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoIgnoreElements.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSourceFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoMapFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelMergeSequential.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/concurrent/SpscLinkedArrayQueue.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxFilterFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxLogFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFromFluxOperator.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDeferContextual.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxName.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxFirstWithValue.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxLog.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSampleTimeout.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDoOnEach.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxPublishMulticast.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxConcatArray.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDefaultIfEmpty.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelLift.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFlatMapMany.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-

jar/reactor/core/publisher/SourceProducer.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/Logger.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/DelegateProcessor.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCompletionStage.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSkip.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/BlockingMonoSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDematerialize.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxTimeout.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoRepeatPredicate.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/BaseSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoTakeUntilOther.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxTake.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxFilter.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelConcatMap.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDoFirst.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSourceMono.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoPeekFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoNever.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoUsing.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxPublishOn.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxOperator.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelFluxName.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSingleCallable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxPeekFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelGroup.java

* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/NextProcessor.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxArray.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCount.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/ImmediateScheduler.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoLift.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoOnAssembly.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxTakeLast.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/NonBlocking.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/SingleScheduler.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/function/Tuple8.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSkipLast.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxContextWrite.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoEmpty.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxGroupBy.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDefaultIfEmpty.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/BlockingIterable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxConcatMap.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxIterable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxOnBackpressureLatest.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoMaterialize.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/Fuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxRepeat.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/function/Tuples.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/Metrics.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxErrorOnRequest.java

* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/DrainUtils.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoToCompletableFuture.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/function/Tuple2.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDefer.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoLiftFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoNext.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSubscribeOnCallable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxWindowPredicate.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDistinctFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxTakeLastOne.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxAutoConnectFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/Hooks.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ConnectableFluxOnAssembly.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSubscribeOn.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoAll.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDoOnEachFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCurrentContext.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoHasElement.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxOnAssembly.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoHasElements.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoElapsed.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxIndex.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoPublishOn.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoReduce.java

* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSource.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxWindowWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSubscribeOnCallable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelFlatMap.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxErrorSupplied.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSource.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxMergeSequential.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoIgnoreElement.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSubscribeOnValue.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFilterFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/OptimizableOperator.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxDistinctUntilChanged.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/BlockingSingleSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/InnerConsumer.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoTimed.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelLiftFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/ElasticScheduler.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxError.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxScan.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxUsing.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/Signal.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelReduceSeed.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSourceFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSubscribeOn.java

* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxCombineLatest.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelMergeOrdered.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxPublish.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxBufferPredicate.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxConcatIterable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ConnectableFluxHide.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoFirstWithValue.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelRunOn.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoDelaySubscription.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCancelOn.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoSingleMono.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxRepeatPredicate.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelMap.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxSkipUntilOther.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxTakeWhile.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2020-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/DelegatingScheduledExecutorService.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SinkEmptySerialized.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SinkEmptyMulticast.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SinkManyBestEffort.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxConcatMapNoPrefetch.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/retry/ImmutableRetrySignal.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/UnicastManySinkNoBackpressure.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/retry/Retry.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SinkOneSerialized.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/InternalOneSink.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/retry/RetryBackoffSpec.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/retry/RetrySpec.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/Timed.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/InternalManySink.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/SinkManySerialized.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxTimed.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/InternalEmptySink.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2018-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxMergeComparing.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/OnNextFailureStrategy.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoMetricsFuseable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/Traces.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/QueueDrainSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/concurrent/MpscLinkedQueue.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxMetrics.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/SchedulerMetricDecorator.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/ReactorThreadFactory.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoMetrics.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxMetricsFuseable.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2015-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-

jar/reactor/core/publisher/GroupedFlux.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/Disposable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/scheduler/ExecutorServiceWorker.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ParallelArraySource.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoRunnable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxOnBackpressureBufferStrategy.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/BufferOverflowStrategy.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/BlockingLastSubscriber.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxOnBackpressureBuffer.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoJust.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/InnerOperator.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/ContextHolder.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/FluxJust.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2017 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/annotation/Nullable.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/annotation/NonNull.java

* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/util/annotation/NonNullApi.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCacheInvalidateWhen.java
* /opt/cola/permits/1519137074_1671725129.5872471/0/reactor-core-3-4-21-sources-jar/reactor/core/publisher/MonoCacheInvalidateIf.java

1.179 netty-transport-native-unix-common 4.1.79.Final

1.179.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2021 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.

*/

Found in path(s):
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/DomainDatagramChannel.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/SegmentedDatagramPacket.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/DomainDatagramSocketAddress.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/DomainDatagramChannelConfig.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/DomainDatagramPacket.java
No license file was found, but licenses were detected in source scan.

<!--
~ Copyright 2016 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~   https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->

Found in path(s):
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/META-INF/maven/io.netty/netty-transport-native-unix-common/pom.xml
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the

* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/UnixChannelUtil.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_jni.h
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix.h
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix.c
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2022 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.

*/

Found in path(s):
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/GenericUnixChannelOption.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/RawUnixChannelOption.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/IntegerUnixChannelOption.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2018 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_buffer.h
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_buffer.c
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/PreferredDirectByteBufAllocator.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/Buffer.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/NativeInetAddress.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/Socket.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_errors.c
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/FileDescriptor.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_socket.h
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/DomainSocketAddress.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/DatagramSocketAddress.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/DomainSocketReadMode.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/Errors.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/DomainSocketChannel.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_errors.h
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_filedescriptor.c
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_filedescriptor.h
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_socket.c
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/UnixChannel.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/DomainSocketChannelConfig.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/ServerDomainSocketChannel.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance

* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/package-info.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/IovArray.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/Unix.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/UnixChannelOption.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/Limits.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/SocketWritableByteChannel.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_limits.h
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_util.c
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-

2-jar/io/netty/channel/unix/ErrorsStaticallyReferencedJniMethods.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/LimitsStaticallyReferencedJniMethods.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/io/netty/channel/unix/PeerCredentials.java
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_limits.c
* /opt/cola/permits/1519136880_1671710660.9079401/0/netty-transport-native-unix-common-4-1-79-final-sources-2-jar/netty_unix_util.h

1.180 objenesis 3.2

1.180.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.181 asm 9.2

1.181.1 Available under license :

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/signature/package.html
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/package.html
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//   notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//   notice, this list of conditions and the following disclaimer in the
//   documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//   this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/RecordComponentWriter.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/ClassTooLargeException.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/FieldVisitor.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Frame.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Context.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/MethodTooLargeException.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Label.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Type.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/ClassWriter.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/MethodWriter.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/ClassReader.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/signature/SignatureWriter.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/RecordComponentVisitor.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/ClassVisitor.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/SymbolTable.java

* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Handle.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/AnnotationVisitor.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/ModuleWriter.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/FieldWriter.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/signature/SignatureReader.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/MethodVisitor.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Edge.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/ByteVector.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/signature/SignatureVisitor.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Handler.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Symbol.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/TypePath.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Constants.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/ConstantDynamic.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Attribute.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/AnnotationWriter.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/Opcodes.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/CurrentFrame.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/ModuleVisitor.java
* /opt/cola/permits/1221704254_1635458831.0/0/asm-9-2-sources-2-jar/org/objectweb/asm/TypeReference.java

1.182 fabric8-::-kubernetes-model-::-coordination 5.12.1

1.182.1 Available under license :

No license file was found, but licenses were detected in source scan.

/**
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340815642_1654860894.6408663/0/kubernetes-model-coordination-5-12-1-sources-jar/io/fabric8/kubernetes/api/model/coordination/v1/LeaseSpec.java
No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

       http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815642_1654860894.6408663/0/kubernetes-model-coordination-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-model-coordination/pom.xml
No license file was found, but licenses were detected in source scan.

*
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1340815642_1654860894.6408663/0/kubernetes-model-coordination-5-12-1-sources-jar/manifest.vm

1.183 h2 2.1.214

1.183.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: Jan Kotek
*/

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/CacheSecondLevel.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0, and the
* EPL 1.0 (https://h2database.com/html/license.html). Initial Developer: H2
* Group
*/

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/LobStorageRemoteInputStream.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/ConditionAndOrN.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcDatabaseMetaDataBackwardsCompat.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcConnection.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueLob.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: Lazarev Nikita <lazarevn@ispras.ru>
*/

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueJson.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
*

* This code is based on the LZF algorithm from Marc Lehmann. It is a
* re-implementation of the C code:
* http://cvs.schmorp.de/liblzf/lzf_c.c?view=markup
* http://cvs.schmorp.de/liblzf/lzf_d.c?view=markup
*
* According to a mail from Marc Lehmann, it's OK to use his algorithm:
* Date: 2010-07-15 15:57
* Subject: Re: Question about LZF licensing
* ...
* The algorithm is not copyrighted (and cannot be copyrighted afaik) - as long
* as you wrote everything yourself, without copying my code, that's just fine
* (looking is of course fine too).
* ...
*
* Still I would like to keep his copyright info:
*
* Copyright (c) 2000-2005 Marc Alexander Lehmann <schmorp@schmorp.de>
* Copyright (c) 2005 Oren J. Maurice <oymaurice@hazorea.org.il>
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:
*
*   1. Redistributions of source code must retain the above copyright notice,
*   this list of conditions and the following disclaimer.
*
*   2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
*
*   3. The name of the author may not be used to endorse or promote products
*   derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
* MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
* EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS;
* OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
* OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF
* ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/compress/CompressLZF.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: Christian d'Heureuse, www.source-code.biz
*
* This class is multi-licensed under LGPL, MPL 2.0, and EPL 1.0.
*
* This module is free software: you can redistribute it and/or
* modify it under the terms of the GNU Lesser General Public
* License as published by the Free Software Foundation, either
* version 3 of the License, or (at your option) any later version.
* See https://www.gnu.org/licenses/lgpl-3.0.html
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied
* warranty of MERCHANTABILITY or FITNESS FOR A
* PARTICULAR PURPOSE. See the GNU Lesser General Public
* License for more details.
*/

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbcx/JdbcConnectionPool.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: Daniel Gredler
*/

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/ToCharFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/ToDateTokenizer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/ToDateParser.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: Alessandro Ventura
*/

Found in path(s):

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/CredentialsValidator.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/ConfigProperties.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/HasConfigProperties.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/DefaultAuthenticator.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/AuthenticatorFactory.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/H2AuthConfig.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/UserToRolesMapper.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/UserToRolesMapperConfig.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/AuthenticationException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/impl/AssignRealmNameRole.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/Configurable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/impl/StaticRolesMapper.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/AuthenticationInfo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/impl/StaticUserCredentialsValidator.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/H2AuthConfigXml.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/RealmConfig.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/AuthConfigException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/PropertyConfig.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/impl/LdapCredentialsValidator.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/Authenticator.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/auth/impl/JaasCredentialsValidator.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0, and the
* EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: H2 Group

* Iso8601: Initial Developer: Robert Rathsack (firstName dot lastName at gmx
* dot de)
*/

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/DateTimeUtils.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: H2 Group
* Iso8601:
* Initial Developer: Robert Rathsack (firstName dot lastName at gmx dot de)
*/

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/CloseWatcher.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: H2 Group
*/

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/value/ValueToObjectConverter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/function/MathFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/value/lob/LobDataInMemory.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/table/TableValueConstructorTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/dml/SetSessionCharacteristics.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/function/SignalFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/result/ResultWithPaddedStrings.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/mvstore/db/NullValueDataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/mvstore/db/MVDelegateIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/index/VirtualConstructedTableIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-

jar/org/h2/constraint/ConstraintUnique.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/Right.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/jdbc/JdbcParameterMetaData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/mvstore/rtree/DefaultSpatial.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/ddl/AlterDomainRenameConstraint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/analysis/PartitionData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/mvstore/db/MVTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/function/SessionControlFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/value/lob/LobDataDatabase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/engine/GeneratedKeysMode.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/mode/FunctionsDB2Derby.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/util/json/JSONItemType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/LocalResult.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/condition/Condition.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/condition/CompareLike.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/rec/FileRec.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/DerivedTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/index/VirtualTableIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/ddl/CreateConstant.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/util/geometry/JTSUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/Aggregate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/ExpressionWithFlags.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/mvstore/db/MVSecondaryIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/ddl/CreateView.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/util/json/JSONStringSource.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/condition/Comparison.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/value/CharsetCollator.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/VirtualTableCursor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/SmallLRUCache.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/SHA256.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Variable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/CurrentDateTimeValueFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueEnumBase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/GrantRevoke.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/FunctionsMSSQLServer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONValueTarget.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterTableRenameConstraint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/MVSortedTempResult.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONNull.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/Update.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/RuleHead.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jmx/DatabaseInfo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/Typed.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/CountingReaderInputStream.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/meta/DatabaseMetaServer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/constraint/ConstraintReferential.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/SearchedCase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/pg/PgServerThread.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/RootReference.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/LinkedCursor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/MetaIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/LinkedIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/VersionedValueCommitted.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/MemoryEstimator.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/IsolationLevel.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/FunctionAlias.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/analysis/DataAnalysisOperation.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/InformationSchemaTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLNonTransientConnectionException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/cache/CacheLongKeyLIRS.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/RecoverTester.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/TableBase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/JavaObjectSerializer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/analysis/WindowFunctionType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/analysis/Window.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jmx/DatabaseInfoMBean.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/lob/LobDataFile.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/DualCursor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/CacheLRU.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/ConditionLocalAndGlobal.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/niomapped/FilePathNioMapped.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/IndexHints.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/RangeTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/web/DbStarter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/SimpleRowValue.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueToObjectConverter2.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/TableEngine.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/RangeInputStream.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/context/DbColumn.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLInvalidAuthorizationSpecException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/SysInfoFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/niomem/FilePathNioMemLZF.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/Server.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcConnectionBackwardsCompat.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/CreateSchema.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/ExecuteImmediate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/Constant.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/geometry/GeoJsonUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropTrigger.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/IndexCondition.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ExtTypeInfoEnum.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/ConditionNot.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/TriggerAdapter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/StreamStore.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/TruncateTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueInteger.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/Insert.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/Restore.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropRole.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbcx/JdbcXid.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/SetTypes.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/SequenceValue.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateDataBinarySet.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropSynonym.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/type/StringDataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/fulltext/FullText.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/ByteStack.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropConstant.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLSyntaxErrorException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueTinyint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/query/SelectUnion.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/constraint/ConstraintCheck.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Subquery.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/message/DbException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/SecureFileStore.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/SortedProperties.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/RunScriptCommand.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLDataException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/CardinalityExpression.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/FileStoreOutputStream.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/MVInDoubtTransaction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/query/SelectListColumnResolver.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/MathFunction1.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/meta/DatabaseMetaLocal.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Alias.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONValue.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcPreparedStatement.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/FileBaseDefault.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/table/LinkSchemaFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Operation1_2.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/analysis/WindowFrameExclusion.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/QueryExpressionIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/ScriptBase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/H2Type.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/Transfer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/SchemaCommand.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/AbbaLockingDetector.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/TriggerObject.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/Bits.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-

jar/org/h2/expression/function/SubstringFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/CacheObject.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/value/ValueTimestampTimeZone.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/condition/ExistsPredicate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/Table.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/Command.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/function/TrimFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/Snapshot.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/ddl/CreateDomain.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/constraint/Constraint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/function/CryptFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/GUIConsole.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/ExpressionVisitor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/dml/NoOperation.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/Rule.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/result/ResultInterface.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/jdbcx/JdbcXAConnection.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/RuleElement.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/Profiler.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/SpatialIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/aggregate/AbstractAggregate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/Engine.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/aggregate/AggregateDataAvg.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/IndexColumn.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/condition/ConditionInQuery.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/ddl/CreateTrigger.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueSmallint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/MergedResult.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/condition/ConditionInConstantSet.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/Chunk.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueDecfloat.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/value/lob/LobDataFetchOnDemand.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcBlob.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/type/ObjectDataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/message/Trace.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/UserBuilder.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/IndexType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/compress/CompressNo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/MetaRecord.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/CastSpecification.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/CommitDecisionMaker.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/Cache.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/SimpleResult.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/DefaultNullOrdering.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/disk/FilePathDisk.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/LobStorageMap.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/FunctionsPostgreSQL.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/mem/FileMemData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/TransactionCommand.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/RuleOptional.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateDataCount.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/Token.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLIntegrityConstraintViolationException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterSchemaRename.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/RangeReader.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/ParameterInterface.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/ConcatenationOperation.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropAggregate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/PrepareProcedure.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-

jar/org/h2/expression/function/TableInfoFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/query/Query.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONArray.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/analysis/WindowFrame.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/UserDefinedFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/SetComment.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/ColumnTemplate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONObject.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/NetworkConnectionInfo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/XMLFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/type/DataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/fulltext/IndexInfo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLTransientException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueBigint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropUser.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/VirtualTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/Script.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JsonConstructorUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/TableType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/retry/FileRetryOnInterrupt.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/ParserUtil.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/Database.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DefineCommand.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/ArrayElementReference.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/geometry/EWKTUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/FunctionInfo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/type/ByteArrayDataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcStatement.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/FilePathWrapper.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/SmallMap.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/SourceCompiler.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/Recover.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/IntervalOperation.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/FileUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueVarchar.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/analysis/WindowFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/Row.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/Cursor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/web/WebApp.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueTimestamp.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/MVMap.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/MVStore.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/message/TraceSystem.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/lob/LobData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/Store.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/TableLink.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateDataStdVar.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/Sequence.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/PredicateWithSubquery.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/RegexpFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/ValueExpression.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterDomainAddConstraint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/NullPredicate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/split/FileSplit.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/TxDecisionMaker.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/OsgiDataSourceFactory.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/ConditionAndOr.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateDataCorr.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/RangeCursor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ExtTypeInfoRow.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/CreateAggregate.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/SchemaOwnerCommand.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/UserAggregate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/FakeFileChannel.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/ConcatFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/MVStoreTool.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/RowDataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterView.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueVarcharIgnoreCase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLNonTransientException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/DBObjectFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/FieldReference.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbcx/JdbcDataSourceFactory.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/FileStoreInputStream.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/CommandInterface.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/Recorder.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/ExpressionWithVariableParameters.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/SoftValuesHashMap.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/compress/LZFOutputStream.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/Shell.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/async/FilePathAsync.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/CreateTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/meta/DatabaseMeta.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/CastDataProvider.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/GeneratedColumnResolver.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/JdbcDriverBackwardsCompat.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/AbbaDetector.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropDatabase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueChar.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/DbSettings.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Operation2.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/TempFileDeleter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/ModeFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/ConvertTraceFile.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/SchemaObject.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/BnfVisitor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueGeometry.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueUuid.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/CompareMode.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/message/TraceWriter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/ResultColumn.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/Call.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/MetaCursor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/CacheWriter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONValidationTarget.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterIndexRename.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/MVIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/IntArray.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/zip/FilePathZip.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/type/MetaType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/query/AllColumnsForPlan.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/LongDataCounter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/ErrorCode.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/TypedValueExpression.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Expression.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropFunctionAlias.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/RowFactory.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/FunctionsLegacy.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/niomapped/FileNioMapped.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/encrypt/FileEncrypt.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/SimpleColumnInfo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/MVSpatialIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/AggregateFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueCollectionBase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DeallocateProcedure.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/encrypt/FilePathEncrypt.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/LobStorageInterface.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/RightOwner.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/FileLockMethod.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/Procedure.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/StringFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/PlanItem.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/table/TableFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/Cursor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueTime.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/BooleanTest.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/FunctionN.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/CreateFunctionAlias.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateDataCovar.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/Driver.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/VersionedValueUncommitted.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/Function1.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/web/ConnectionInfo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterTableDropConstraint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropSchema.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/meta/DatabaseMetaLegacy.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueDouble.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueBoolean.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/CommandRemote.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/ConditionIn.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/CreateLinkedTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbcx/JdbcDataSource.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/BlockCipher.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/Sentence.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/fulltext/FullTextSettings.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/IndexCursor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/CreateCluster.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/context/DbTableOrView.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/DataChangeStatement.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueInterval.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/Fog.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/DelayedDatabaseCloser.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/CoalesceFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/CompressFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/DataHandler.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/DebuggingThreadLocal.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/compress/CompressDeflate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/CursorPos.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONBoolean.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/CreateSequence.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/SysProperties.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/Tool.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/Prepared.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/Comment.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/analysis/WindowFrameBound.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/ScriptReader.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/StringFunction1.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/CompareModeIcu4J.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLXML.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/CurrentGeneralValueSpecification.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/PgCatalogSchema.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/ScriptCommand.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropDomain.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/Merge.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLFeatureNotSupportedException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLTransactionRollbackException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/ChangeFileEncryption.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/TableFilter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/IntervalUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/geometry/EWKBUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONBytesSource.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/rtree/MVRTreeMap.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/ResultWithGeneratedKeys.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/query/QueryOrderBy.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/split/FilePathSplit.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterSequence.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueEnum.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/Transaction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/Help.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/web/JakartaWebServlet.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/DataUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/XTEA.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcBatchUpdateException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-

jar/org/h2/tools/SimpleRowSource.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterDomainRename.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcLob.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/CompatibilityDateTimeValueFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/CreateUser.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterTableAlterColumn.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/NullIfFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/FetchedResult.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/FileBase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/SetFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueJavaObject.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/ListaggArguments.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/context/DbSchema.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/MultiDimension.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/niomem/FileNioMemData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/table/CSVReadFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/MathUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueRow.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/Function2.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSavepoint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/query/Select.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/CSVWriteFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/MVStoreException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueStringBase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Operation1.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/Set.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/CipherFactory.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueTimeTimeZone.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/TcpServerThread.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/Backup.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/TcpServer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/ResultTarget.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/MVPlainTempResult.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueDate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/FreeSpaceBitSet.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/web/WebThread.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONTextSource.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/Page.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/FunctionTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/Utils10.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Rownum.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterDomain.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/JavaAggregate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/Function1_2.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/geometry/GeometryUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/UnaryOperation.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/VersionedValueType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/TypeInfo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/rtree/Spatial.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/niomem/FilePathNioMem.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/MetaSchema.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/CompareModeDefault.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/LengthFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/VirtualConstructedTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/OperationN.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/Analyze.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/meta/DatabaseMetaRemote.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-

jar/org/h2/command/ddl/CreateRole.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/web/WebServer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/context/DbContextRule.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/FileStore.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/TimeZoneProvider.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueNumeric.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/web/PageParser.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/ConditionInParameter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/Plan.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/CommandWithColumns.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/FileStore.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/MergeUsing.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/rec/FilePathRec.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/TableLinkConnection.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/CreateIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/fulltext/FullTextLucene.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jmx/DocumentedMBean.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/ResultExternal.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/InformationSchema.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONByteArrayTarget.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/User.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/CompatibilitySequenceValueFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/DefaultRow.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/JdbcUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/StringFunction2.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/Constants.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbcx/JdbcConnectionPoolBackwardsCompat.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/SessionRemote.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONStringTarget.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ExtTypeInfoNumeric.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/CompressTool.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/SimpleResultSet.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/IsJsonPredicate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/CreateSynonymData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/PgCatalogTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropView.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/SoundexFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/ThreadDeadlockDetector.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/HasSQL.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/Index.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/Sparse.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/SingleRowCursor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/SpatialKey.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/web/WebServlet.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterDomainExpressions.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/Explain.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/LazyResult.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/BitFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/pg/PgServer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/type/LongDataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/Setting.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/FileFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/ValueDataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/CaseInsensitiveMap.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/ExpressionList.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/mem/FilePathMem.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-

jar/org/h2/command/dml/Delete.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/constraint/ConstraintActionType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/SettingsBase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/CaseInsensitiveConcurrentMap.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcCallableStatement.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Wildcard.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/DataTypeSQLFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/DropSequence.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/RandFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Format.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/MemoryUnmapper.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueBinary.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/Tokenizer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/DatabaseEventListener.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/type/BasicDataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/ShutdownHandler.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/Task.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/OnExitDatabaseCloser.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/DualIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/Value.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcClob.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateDataDistinctWithCounts.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/UpdatableRow.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/DbObject.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateDataCollecting.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONTarget.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/Trigger.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONValidationTargetWithoutUniqueKeys.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/ConnectionInfo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterTableRenameColumn.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-

jar/org/h2/command/ddl/AlterUser.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/FileLister.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/ddl/DropIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/condition/BetweenPredicate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/query/SelectGroups.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/Parameter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/RuleFixed.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/FilePath.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/store/fs/niomem/FileNioMem.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/value/VersionedValue.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/server/web/WebSession.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/ddl/CreateSynonym.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/TableView.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/value/ValueVarbinary.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/SortOrder.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/Console.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/Regclass.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/function/ArrayFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/jdbc/meta/DatabaseMetaLocalBase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/mvstore/OffHeapStore.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/compress/LZFInputStream.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/bnf/context/DbProcedure.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/mvstore/rtree/SpatialDataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/RunScript.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/expression/BinaryOperation.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/store/fs/mem/FilePathMemLZF.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/engine/QueryStatisticsData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-
jar/org/h2/command/ddl/CreateTableData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-

jar/org/h2/mvstore/tx/TransactionStore.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/LegacyDateTimeUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/message/TraceObject.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/Role.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/CommandList.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONNumber.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONValidationTargetWithUniqueKeys.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/JavaFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/Bnf.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/DataReader.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/ExecuteProcedure.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/FilteredDataChangeStatement.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/DataChangeDeltaTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/RangeIndex.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterTableAddConstraint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/Interval.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/AES.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/DataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/Operation0.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/RuleExtension.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcDatabaseMetaData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/table/JavaTableFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/ColumnResolver.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/cache/FilePathCache.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/StringUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/RollbackDecisionMaker.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/CommandWithValues.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/Column.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-

jar/org/h2/expression/aggregate/Percentile.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/DbDriverActivator.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/CacheHead.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/BuiltinFunctions.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/VersionedBitSet.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/RuleList.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/Domain.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/LobStorageFrontend.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbcx/JdbcDataSourceBackwardsCompat.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/CommandContainer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/context/DbContents.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/web/JakartaDbStarter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/SimpleCase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/TableSynonym.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/UniquePredicate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueReal.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/FileChannelInputStream.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterTableRename.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/constraint/ConstraintDomain.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/MVTempResult.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/constraint/DomainColumnResolver.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueBigDecimalBase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/retry/FilePathRetryOnInterrupt.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueBlob.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/FunctionsOracle.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/ResultRemote.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/db/MVPrimaryIndex.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/AlterTableSet.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateDataDefault.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/InformationSchemaTableLegacy.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueNull.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/TruncateValueFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ExtTypeInfo.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/SetClauseList.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/AggregateDataEnvelope.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/IOUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/AlterDomainDropConstraint.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcArray.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/QueryExpressionTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/ExpressionColumn.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/Utils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/DualTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/table/ArrayTableFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/SimplePredicate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/compress/Compressor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/analysis/WindowFrameUnits.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/message/TraceWriterAdapter.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/zip/FileZip.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/TransactionMap.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueBytesBase.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/ParameterRemote.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/CacheTQ.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/JsonConstructorFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/Session.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/dml/BackupCommand.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/DeleteDbFiles.java

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/DateTimeFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/SessionLocal.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/async/FileAsync.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/table/MetaTable.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/query/TableValueConstructor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/type/StatefulDataType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/FileLock.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcStatementBackwardsCompat.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/mem/FileMem.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/NamedExpression.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/index/QueryExpressionCursor.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/Function0_1.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcSQLTimeoutException.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ExtTypeInfoGeometry.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/ArrayConstructorByQuery.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/bnf/RuleRepeat.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/OnDuplicateKeyValues.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/result/SearchRow.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/engine/Mode.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/CompatibilityDatePlusTimeOperation.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/MathFunction2.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/Csv.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/NetUtils.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/WriteBuffer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcResultSetMetaData.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/fs/encrypt/XTS.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/condition/TypePredicate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/json/JSONString.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/query/Optimizer.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-

jar/org/h2/expression/DomainValueExpression.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/analysis/WindowFrameBoundType.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/api/IntervalQualifier.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/DayMonthNameFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueArray.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/value/ValueClob.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/ddl/SequenceOptions.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/InDoubtTransaction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/tools/Upgrade.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/HashFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/aggregate/Aggregate.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/function/DateTimeFormatFunction.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/security/SHA3.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/jdbc/JdbcResultSet.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mvstore/tx/Record.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/server/Service.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/schema/Schema.java
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/expression/TimeZoneOperation.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: H2 Group
*/
/**
   * Converts a value to a LocalDate.
   *
   * This method should only be called from Java 8 or later version.
   *
   * @param value
   *            the value to convert
   * @param provider
   *            the cast information provider
   * @return the LocalDate
   */
```

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/JSR310Utils.java

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: H2 Group
*
* The variable size number format code is a port from SQLite,
* but stored in reverse order (least significant bits in the first byte).
*/
```

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/store/Data.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: Jason Brittain (jason.brittain at gmail.com)
*/
```

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/mode/FunctionsMySQL.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: H2 Group
*
* According to a mail from Alan Tucker to Chris H Miller from IBM,
* the algorithm is in the public domain:
*
* Date: 2010-07-15 15:57
* Subject: Re: Applied Combinatorics Code
*
* Chris,
* The combinatorics algorithms in my textbook are all not under patent
* or copyright. They are as much in the public domain as the solution to any
* common question in an undergraduate mathematics course, e.g., in my
* combinatorics course, the solution to the problem of how many arrangements
* there are of the letters in the word MATHEMATICS. I appreciate your due
* diligence.
* -Alan
*/
```

Found in path(s):

* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/util/Permutations.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2004-2022 H2 Group. Multiple-Licensed under the MPL 2.0,
* and the EPL 1.0 (https://h2database.com/html/license.html).
* Initial Developer: H2 Group
*
* Nicolas Fortin, Atelier SIG, IRSTV FR CNRS 24888
* Support for the operator "&&" as an alias for SPATIAL_INTERSECTS
*/

Found in path(s):
* /opt/cola/permits/1519136982_1671710531.2417731/0/h2-2-1-214-sources-1-jar/org/h2/command/Parser.java

1.184 opentelemetry 1.14.0

1.184.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute

must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License.

However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.185 latencyutils 2.0.3

1.185.1 Available under license :

PK0P=spring-cloud-starter-netflix-ribbon-2.1.2.RELEASE-sources.jarg4G7C{ F-`DutCD !QG7Dwy>9u:g?{]ic`pqq
_Uh*C5!Jz`p4w{HIE'1z>O<(X
IRq8>H+Y/^*xBZ8}LX1~11_fg7s{{`n`[W_;2|csqpu9x2IETH=a~v<\.0'i>/d1!4F&U5xV~VBEGA?}\O7pk"i)K3)
.#RScs7,_D/_z,:*^+;J~lXE3.<%wt_3[5h]v7S,Dw5bZLI=C3tMDVrJLn9?l,Shthp
'sn}S`o [~ {:8@SI7R$
1f+U7L,`,$Ts,VIow^Mk0p_5S#[n.A **h'"v\NK:qvO< jRT2'eC@Br-KuY.czg]~\&VwiJ6W]Z>0E;zjs~OG_
g+8`?<$O{7@-A'QR+&Uz.+GlL8UODo9wTsV"yCYjI iWmz^'pkJ`pd^V{NjW)x|U#
&V~/Gl{JJ`ySR]gag{M)OL>0av
"nJ9${,VdS} .~E8[MNbW/
kN.Emkhuo~)SY_[3p^Pu#,`c
]Jcoig;C^CxJ6LY!sN~|~B3TKFxAi#g@Z@%glo:L$~A}rno$Opv^$C&=Ut+AU FF[7U-l7{a]Mc
bDcK*B*^HC:>kflV5fZu,Tt<j
JsmXl49y[iu"6\;`UNf&9OnF9i6h;tXhM())wb_UJqiZ_:)1&|,)P(~1wV,^dU_V+m9kmKw\.Lj c:47:@U3^>/-
lu58cxw_\?},|2^Re+ Ao\%IAIeJ02vB;iMASvF}
Ht|a8Q>}g6%2%~k{2##Hs-.7ZY%k*= 0{:_?d:#"8m"sq(DEZR#T?N%kx~IHNVjK
[Kz2n'JI.I8sX)U4vXcej),4{%tEgiF9J DL$j,Onb)cJY)A2N&-dRrE/3D2OLAHRr=*+bWQgbE_UG{Y-
$&6rX@_HA'"k /H(VO9C9O%FZEn6{F BtKDK1zD%u28J<kI+p%HJvQH|q
S.YCm<83=Fn8p $S2EDP$"u4x"ym
ZN!cS<a8,Iyz7KIw{{*xO#B\-*}mD $RGR;)Fp0o0V00B#c=4skVTYo4P@c4gqO*82myazgQ<_/],*Zm_M\e.
nKbdE ^O^yTd/$Vu|=Q HJJ_k0YzH+;+Wb=u`xm
FFbkCs<dJ%_b ht&Sq:)?*TfEe{)E%@mUxT8W}y3 Z8BEFM}tEa7b,gKMUAU'07wm8AQ-+)
D\y:=GlmiN.rh;o$tSqmv}j^m}C>qT?<+^3g5&'Ac%/ei3hq9R = Npo<.Guc{^LmM9^o=!%7I%?^4cd9Hv!-
HN{2+t2Un{&sIBsfw<ov_I<vcfF
$XZE5zAocaf!"RUy6QRabLDkT)
Ce0K_ JW@"cEo,Z~Oru#U0NNyrnq}y56@HXl<_D 6s9/:`EZ91JFR"u:tu}/ksKanG'\X2
Nk>!wn!J'wu+,1iM7 e&V|b
XQU\3},6:BZa852ES$fk\s'F"*w<R47YXv'Z+DXsE1[5Kxcps`aK1dm!p~kSsct,Xo~j$Yn]RYA;D1y-
q/_\UP<ujCQ9/V@OiXj>].+ZV_]Mo/^]?d*NPX Ns![zEvb{P0"|5".?H3Nl}QZ{,4X{.Lvn=<n5t;1
'}7e/@q\
EF2@wr'Hs>|J{h2hI,-oIfU_bF;60EaGd/uV./+WJ(?>ttxuJJhTK^G}Hxwd*DGi"mU(QVN-
NXWUGJaSy3^JSTh#M^ngogsJ k+org]Thcr{+vlDE=kL3>Fr~"VH4~(` P/dSRA\H-mbzdkcf{`-
3amB~eQ4$WKg5F#*=Qx&%W`d
7=
2&Ka L6klnrtKG3'{hOX[D@"6??&$AZKjVBYlN.||6~k]5{W~onI=RjsvXa|YF1+~a+.Yhhj^8 A
({#.\][v!sR'5<;#.^Q~8zUFt#>]l0fvDyTV_v#,x\W
Q;
mmWfw|&+G'iR_VJ: ~B9_P6>oFX!)-u,?zPKDcPK9ZP6spring-cloud-netflix-sidecar-2.2.2.RELEASE-

sources.jarzex]9c1ffff333Cl13cq3_`}GVRTz$A`r}
AW4uo]ZDI #J/- QTU`er,SHrbQ%Qb;p7d
acm6}0o/Fc_H`__S!b0_lu;0K?E~s'8~D-x3sTV\f@jP=AKlgNi
}bIjw:FhPf:HJ/+rTo*WDqnG(LJ$JLj`eE#5_qfCvO}Tdn7L`T'=Brl4;yr.}#Ebsu<Eke@
O'Q#=TQ DKg!foY~i"AaL,#"m2F[C~#pj0~%K'ik^RQpUvpdWQG[Uu5Y)K(RT@yq`$-4a LlHgfBRe&9h-
e1\Fa~i%+\MtW*+Blzi-p7&Q1v'4U{roR d[g~65bSS{-dBU"7)7Enmi{K&6Lum(^)c7KU
a2mMbW+Cs(/0>"~e8@@llpu+%vgNhDxX%
Tf*"$8f'*f(a[Xn@${YQa/=HHHb,WAG
p>a66ff,iz0D)d|_7}i+{TFjCp}WX&k1v-I tst]B <)cri!!;D4+
%!F9\DaP7KF*nn >`
HOm-_/;ta?]w^JzC]`@!VaN=$ODc%*CsI
,wc#HH-
`Z7>3CrZR$qwCm'/p]R!/)r%?Z]Wg^zm{.{!HR@:,D}&k'1cU6/h}(5VtAVeE4vM.<%&<|+93UUlC2t~Ig#,=YiA9H
/-h`p^i8.UIVnF|x-&e<O?dAtHCDj"
c`c<seJMS$*jKr?q&cWAy: XsU/JB!5(yADY)@
e@Cj
J/J)-x53mNxHZi
7rxGwczU'mWl{T4CZd"zpphU`jU
=3X[UtteFhvE+}U+V;]cK?RM(,9W5A(m9jBkQ\~sN'yqg"}j9K)thHJ1NDo$z3n:IiAjh%ACWj;
-mj^
axurii1k/VVtPY]0|<
w\5g.lj-N)V<:y%t4c9 RNS|3(`#dcmlfh`fc{\,W_NTDo(%EQD2i=W_QH"_N%|IC~F+S
zj7i\AvyPrIe)7*nF4VwJuhA2,rS~$q$ywuWX(#B`D==>*/9Nt*1d{OApPd/pdPb@#@8dUd)Y-xQht-
RCU@^5'3NW{A8R f\"0[)=Eu\C]S$aIQ=<v8I80jHOeO.l/Y$E;Eh8%6J\~Q!:9_#h?h@/
4G&mZX3H>v. X98"MBb9 [,\B]tUhMu@EldpXM*;#C[{>n + t;1`<SjpZwNsHeV@o:peR^!@:olx?K]
t
>tc>lLD2C2H+W,D<g5E|N-Ve5 &M<Pc*80SB7Ox)Dg3#/G0f/e5
@ntau1;n{]L
.-&jzSy*q^1G!g&S
q"v{BItz4od^ZrIUq`fbo3GCp^:8x+FRC,uQriTL[xSO2'Z[0+s\4*#>SGzM]l].-">$905E]3b)8Us9y1MjLk}Gq-
W|">%V
Snk87]U$Mv{m1{<\.:z;l7}W8$8UCs&SN4mc.F(@BH$K 8#G#g9GD@T3#?LYoD9
h;GyW%>5pNk;S0iyFbrS-]p[N8Aq]E3d37gkbnOy3 Q_v9\q&2{9`H*,/c]F6dyT"x>Nwb
kP\h,ho7r|7h:bR.)o=U
3jpdpjBS>6;)-k50\|!
dym:=& !#p[hz=^?*w
:bLWc-C (z$SK8OLjJ4:\e\4*U*YlEI,?"u'mKQ)|C7)bg{-sv
2#~YSR "Q]'
LZ]dJ5y;6?6Y)Q+EkkW#TN--Q3dxmg|PIlO0m"ZG%>btV\%?J,MsL^5N31):^f&05CD
8eDgKt"_U#H?E-)YE
4'v61gStI7k$vw^?dQ
g9*B,RYEC0}6I",,NY^yl/W;ji7t1-GdTyK~1mQa.#1X)ayorAMJK}#Q9M~J8$'*+LOIm#
{scW7F/[l0xxB?!(Kt?qpkuh$
o7[4.@rDQ{Bc(U[-b;zw" |($x=0@[xq:b#V65L!lEk<qOuQO%m
gDDwY#x+]C=odD?W8Hp+o'noI&?I#,{"m]z*A4/O>z5J@4Wf,h\twk3D#CM1O&ND{blCPV|P)^Ds'A&lom^TC7
C>e,O]K@.olsoR@cNR~~YKuYV~*$cw2dUCtdap/TzF

LN[pY-bFv9UJ3nbVzd8-3/:dhgqF5)#G/1>fkuV{|Hp&2Vf4NOw%FiU5.0}|4oBc Wn;'&SU2HT@Tg*42`}=mW Yb

;hKs~MU<2 eIbaU:Lpgyc"svh>*bDW2TxiHrcBQU1;&QA`>l8
/KT?pAX;^[hmK y>7??4C[,ecg)`kkifY:
{<,mH>of>Pv9%EH4_VSaqx3S.s;l|ji2?Ci1-IH9N~F)bH8`p`|IKZ(;GB0^xt
RDS23v+iCL#F>Cvkqeq$
gcj%
<Tze"&-7$qm E6Z##.|#'k>|6S9/c3Wu+Mx44Rbl9dJ6)T=8yDl2g
fSL>5M_`<{:3S9Y%zmvRrhdE;)T>[+=|\O_-liv-65 o#wm8'c"7/cGjs(9yO5QkD(!9{/0
{m\NXAtJCi!uc"\<B<IP.DAaY/+;y9MmTJ
i>SPB2CuTR{4K$p8?V 7}"
/]dD;^ c^KI\+}?Y?T0hHDj*J{U$VA&#^N w%*MWa
XIoy9-aR.[1ep&(4nK%EEu_Yk!Y^*TuOy0 &iJ9a|?$
mUJrlf1gd<-T~~tz{^bbxTRZbTv~xNrx~JvHv|BrDV0,CE
+6=^}NrF\Rzjzjt<L7\h6RfX=;XZ2mYt
FrE|]k(!m;D;xtAK9#w<=@*>S$R*^:qD8,~3A=G*Bj(6ePy~R%
0$^I}2]>/"(sE-E:~xNEx)
;YZtk%jTp[XO5*p4,b@n)gye+S[&GF}zD!\\2r5|xR9x^XDvL xH%e(pExM}b-(4jWY#;>.r8&MJy#hD+|sF"2uQ
|H"bX9KZz<6J?~liW* Y_o|O|Fp2NEQ]c$)0D9mIhW{`[h x(v0_a|5Tqn:$k|LxPWO
P~@Ih>Zz~KE9U0sU*}l>qdF@ :%
$kf`phCU#:%aohb5h
sm7iCY/[64::w/aRM.-c4szto@`onq?J?zQANROBttpz8qp5H-g+]
+mVg-@W4MHWnv$]kCxm `qB[8eLiU0mnMy4g&`
^Qx
vd*yC5S~-+ |2 _>$Z5:xz/i.]{p8fM78m}kVO8PHeMGTi:~$0JHcFyaq2&<~
jg4ylkX"+*z9{L
o(B|7{|{h-ICc!s6zJVE.3RK`pDlL.RG6#swX+GBuRg5mlM;[lF}DJ#QZuU=G m\06WaLh3SZh _`SA8va$/c)`6
@h1]C
.!=
"wj\5^XW6|OVo]\};^k3i^
9ws_Iq:D=g '!l'PX8{6Rh8*I|[Tv1^8!c/K1g3Nzj}_pSER"%F_B7F}mr"oS/]PPE:)VFXNk'47p4ZUK4

CA=6JjVHj2Lv[Gy(?"x
Mi0p=M=r)wO[YPUDl% rkX4yMc
k`WR/ABV+qKP?4Z~&`J5F?,/;QjxYg7!"\q8qMk&p-.Lq3aqolASrz#b"Koq3:7Ky0WZ6D+IB\UPU2j;\$LVs^m-
+ETHfF

A@i{nVl\BCS!srY rq25>rgYS3$`?kLjI>{GJwwqS<s9thIWR8_aHD]Zs },[1`uE1H4zG5.2khKPx]/g{?
k1B(,P`^%*Vmo?X,*r:;*$_`~9[B\jr~0a`F1P0ry$^Qa^?VL{ljcdPzKf: v?=oqM~@RTob(aO3i
[GJ,gNWA6La7.QcBz1" 451'cliDC`i|8[Ay0(.8{_)xO`5
4R,r]QNo >UYT<
r? =3LaAnfO"Zo;1mu?.5z+bYhX %
/"7J|C|8qoHjW7Ju~.{ fP 'ad mw7OiG|nsi
u~2r@_3v [Fg@.Nmjino$'+q$}4Wae3r\ZK
[Z6QXf+.s~Y9Wz`e9%)OnI.-VZHM^t:r*ppkMd# Y!Nn6fxs/
a23pT]sO,rpF;&gq|dL)BsZ{mtwciPt=+[5hmU$A@ 9t*bSI?@IbO_!` ?! G/qU'w><0O` aEw`?|Nlu ;~%~
WFwm0G;;_
w>Q6W>
0?yU|fI7W.~w$$8oh~*TPK'&2PK0PDc=spring-cloud-starter-netflix-ribbon-2.1.2.RELEASE-
sources.jarPK9ZP'&26spring-cloud-netflix-sidecar-2.2.2.RELEASE-sources.jarPK=

1.186 spring-cloud-context 3.1.3

1.186.1 Available under license :

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1519137029_1671710564.0767782/0/spring-cloud-context-3-1-3-jar/META-INF/maven/org.springframework.cloud/spring-cloud-context/pom.xml

1.187 jboss-logging 3.4.3.Final

1.187.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,

including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf

of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.188 spring-boot-validation-starter 2.7.2

1.188.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Spring Boot 2.7.2
Copyright (c) 2012-2022 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0 (the "License"). You may not use this product except in compliance with the License.

1.189 spring-security 5.7.2

1.189.1 Available under license :

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate

as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify

the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include

the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
==
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Spring Security distribution. ==
==

The end-user documentation included with a redistribution, if any,
must include the following acknowledgement:

"This product includes software developed by Spring Security
Project (https://www.springframework.org/security)."

Alternately, this acknowledgement may appear in the software itself,
if and wherever such third-party acknowledgements normally appear.

The names "Spring", "Spring Security", "Spring Security System",
"SpringSource", "Acegi", "Acegi Security", "Acegi Security System",
"Acegi" or any derivatives thereof may not be used to endorse or
promote products derived from this software without prior written
permission. For written permission, please contact
ben.alex@springsource.com.

1.190 jackson-bom 2.13.3

1.190.1 Available under license :

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>
<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>

Found in path(s):
* /opt/cola/permits/1343419269_1655288133.0545802/0/jackson-bom-2-13-3-pom-zip/jackson-bom-2.13.3.pom

1.191 opentest4j-opentest4j 1.2.0

1.191.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object

form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a

file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.192 launchdarkly-sdk-for-java 5.7.1

1.192.1 Available under license :

Apache Commons Codec
Copyright 2002-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from http://aspell.net/test/orig/batch0.tab.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

==========

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.
Note that publicsuffixes.gz is compiled from The Public Suffix List:
https://publicsuffix.org/list/public_suffix_list.dat

It is subject to the terms of the Mozilla Public License, v. 2.0:
https://mozilla.org/MPL/2.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.193 checkstyle 9.0.1

1.193.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// This library is free software; you can redistribute it and/or
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
// Lesser General Public License for more details.
// You should have received a copy of the GNU Lesser General Public
/**
* <p>
* Checks that a specified pattern exists, exists less than
* a set number of times, or does not exist in the file.
* </p>
* <p>
* This check combines all the functionality provided by
* <a href="https://checkstyle.org/config_header.html#RegexpHeader">RegexpHeader</a>
* except supplying the regular expression from a file.
* </p>
* <p>
* It differs from them in that it works in multiline mode. Its regular expression
* can span multiple lines and it checks this against the whole file at once.
* The others work in singleline mode. Their single or multiple regular expressions
* can only span one line. They check each of these against each line in the file in turn.
* </p>
* <p>
* <b>Note:</b> Because of the different mode of operation there may be some
* changes in the regular expressions used to achieve a particular end.
* </p>
* <p>
* In multiline mode...
* </p>
* <ul>
* <li>
* {@code ^} means the beginning of a line, as opposed to beginning of the input.
```

```
* </li>
* <li>
* For beginning of the input use {@code \A}.
* </li>
* <li>
* {@code $} means the end of a line, as opposed to the end of the input.
* </li>
* <li>
* For end of input use {@code \Z}.
* </li>
* <li>
* Each line in the file is terminated with a line feed character.
* </li>
* </ul>
* <p>
* <b>Note:</b> Not all regular expression engines are created equal.
* Some provide extra functions that others do not and some elements
* of the syntax may vary. This check makes use of the
* <a href="https://docs.oracle.com/en/java/javase/11/docs/api/java.base/java/util/regex/package-summary.html">
* java.util.regex package</a>; please check its documentation for details
* of how to construct a regular expression to achieve a particular goal.
* </p>
* <p>
* <b>Note:</b> When entering a regular expression as a parameter in
* the XML config file you must also take into account the XML rules. e.g.
* if you want to match a &lt; symbol you need to enter &amp;lt;.
* The regular expression should be entered on one line.
* </p>
* <ul>
* <li>
* Property {@code format} - Specify the pattern to match against.
* Type is {@code java.util.regex.Pattern}.
* Default value is {@code "^$"}.
* </li>
* <li>
* Property {@code message} - Specify message which is used to notify about
* violations, if empty then the default (hard-coded) message is used.
* Type is {@code java.lang.String}.
* Default value is {@code null}.
* </li>
* <li>
* Property {@code illegalPattern} - Control whether the pattern is required or illegal.
* Type is {@code boolean}.
* Default value is {@code false}.
* </li>
* <li>
* Property {@code duplicateLimit} - Control whether to check for duplicates
* of a required pattern, any negative value means no checking for duplicates,
```

```
* any positive value is used as the maximum number of allowed duplicates,
* if the limit is exceeded violations will be logged.
* Type is {@code int}.
* Default value is {@code 0}.
* </li>
* <li>
* Property {@code errorLimit} - Specify the maximum number of violations before
* the check will abort.
* Type is {@code int}.
* Default value is {@code 100}.
* </li>
* <li>
* Property {@code ignoreComments} - Control whether to ignore matches found within comments.
* Type is {@code boolean}.
* Default value is {@code false}.
* </li>
* </ul>
* <p>
* To configure the check:
* </p>
* <p>
* The following examples are mainly copied from the other 3 checks mentioned above,
* to show how the same results can be achieved using this check in place of them.
* </p>
* <p>
* <b>To use like Required Regexp check:</b>
* </p>
* <p>
* An example of how to configure the check to make sure a copyright statement
* is included in the file:
* </p>
* <p>
* The statement.
* </p>
* <pre>
* // This code is copyrighted
* </pre>
* <p>
* The check.
* </p>
* <pre>
* &lt;module name="Regexp"&gt;
*   &lt;property name="format" value="// This code is copyrighted"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* Your statement may be multiline.
* </p>
```

```
* <pre>
* // This code is copyrighted
* // (c) MyCompany
* </pre>
* <p>
* Then the check would be.
* </p>
* <pre>
* &lt;module name="Regexp"&gt;
*   &lt;property name="format" value="// This code is copyrighted\n// \(c\) MyCompany"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* <b>Note:</b> To search for parentheses () in a regular expression you must
* escape them like \(\). This is required by the regexp engine, otherwise it will
* think they are special instruction characters.
* </p>
* <p>
* And to make sure it appears only once:
* </p>
* <pre>
* &lt;module name="Regexp"&gt;
*   &lt;property name="format" value="// This code is copyrighted\n// \(c\) MyCompany"/&gt;
*   &lt;property name="duplicateLimit" value="0"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* It can also be useful to attach a meaningful message to the check:
* </p>
* <pre>
* &lt;module name="Regexp"&gt;
*   &lt;property name="format" value="// This code is copyrighted\n// \(c\) MyCompany"/&gt;
*   &lt;property name="message" value="Copyright"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* <b>To use like illegal regexp check:</b>
* </p>
* <p>
* An example of how to configure the check to make sure there are no calls to
* {@code System.out.println}:
* </p>
* <pre>
* &lt;module name="Regexp"&gt;
*   &lt;!-- . matches any character, so we need to escape it and use \. to match dots. --&gt;
*   &lt;property name="format" value="System\.out\.println"/&gt;
*   &lt;property name="illegalPattern" value="true"/&gt;
* &lt;/module&gt;
```

```
* </pre>
* <p>
* You may want to make the above check ignore comments, like this:
* </p>
* <pre>
* &lt;module name="Regexp"&gt;
*   &lt;property name="format" value="System\.out\.println"/&gt;
*   &lt;property name="illegalPattern" value="true"/&gt;
*   &lt;property name="ignoreComments" value="true"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* An example of how to configure the check to find trailing whitespace at the end of a line:
* </p>
* <pre>
* &lt;module name="Regexp"&gt;
*   &lt;property name="format" value="[ \t]+$"/&gt;
*   &lt;property name="illegalPattern" value="true"/&gt;
*   &lt;property name="message" value="Trailing whitespace"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* An example of how to configure the check to find case-insensitive occurrences of "debug":
* </p>
* <pre>
* &lt;module name="Regexp"&gt;
*   &lt;property name="format" value="(?i)debug"/&gt;
*   &lt;property name="illegalPattern" value="true"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* <b>Note:</b> The (?i) at the beginning of the regular expression tells the
* regexp engine to ignore the case.
* </p>
* <p>
* There is also a feature to limit the number of violations reported.
* When the limit is reached the check aborts with a message reporting that
* the limit has been reached. The default limit setting is 100,
* but this can be change as shown in the following example.
* </p>
* <pre>
* &lt;module name="Regexp"&gt;
*   &lt;property name="format" value="(?i)debug"/&gt;
*   &lt;property name="illegalPattern" value="true"/&gt;
*   &lt;property name="errorLimit" value="1000"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
```

```
* <b>To use like <a href="https://checkstyle.org/config_header.html#RegexpHeader">
* RegexpHeader</a>:</b>
* </p>
* <p>
* To configure the check to verify that each file starts with the following multiline header.
* </p>
* <p>
* Note the following:
* </p>
* <ul>
* <li>
* \A means the start of the file.
* </li>
* <li>
* The date can be any 4 digit number.
* </li>
* </ul>
* <pre>
* // Copyright (C) 2004 MyCompany
* // All rights reserved
* </pre>
* <pre>
* &lt;module name="Regexp"&gt;
*   &lt;property
*     name="format"
*     value="\A// Copyright \(C\) \d\d\d\d MyCompany\n// All rights reserved"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* A more complex example. Note how the import and javadoc multilines are handled,
* there can be any number of them.
* </p>
* <pre>
* ////////////////////////////////////////////////////////////////////
* // checkstyle:
* // Checks Java source code for adherence to a set of rules.
* // Copyright (C) 2004  Oliver Burn
* // Last modification by $Author A.N.Other$
* ////////////////////////////////////////////////////////////////////
*
* package com.puppycrawl.checkstyle;
*
* import java.util.thing1;
* import java.util.thing2;
* import java.util.thing3;
*
* &#47;**
* * javadoc line 1
```

* * javadoc line 2
* * javadoc line 3
* */
* </pre>
* <pre>
* <module name="Regexp">
*   <property
*   name="format"
*   value="\A/{71}\n// checkstyle:\n// Checks Java source code for
*   adherence to a set of rules\.\n// Copyright \(C\) \d\d\d\d Oliver Burn\n
*   // Last modification by \$Author.*\$\n/{71}\n\npackage [\w\.]*;\n\n
*   (import [\w\.]*;\n)*\n/**\n(*[^/]*\n)* */"/>
* </module>
* </pre>
*
* More examples:
*
*
* The next 2 examples deal with the following example Java source file:
*
* <pre>
* /*
* * PID.java
* *
* * Copyright (c) 2001 ACME
* * 123 Some St.
* * Somewhere.
* *
* * This software is the confidential and proprietary information of ACME.
* * ("Confidential Information"). You shall not disclose such
* * Confidential Information and shall use it only in accordance with
* * the terms of the license agreement you entered into with ACME.
* *
* * $Log: config_misc.xml,v $
* * Revision 1.7 2007/01/16 12:16:35 oburn
* * Removing all reference to mailing lists
* *
* * Revision 1.6 2005/12/25 16:13:10 o_sukhodolsky
* * Fix for rfe 1248106 (TYPECAST is now accepted by NoWhitespaceAfter)
* *
* * Fix for rfe 953266 (thanks to Paul Guyot (pguyot) for submitting patch)
* * IllegalType can be configured to accept some abstract classes which
* * matches to regexp of illegal type names (property legalAbstractClassNames)
* *
* * TrailingComment now can be configured to accept some trailing comments
* * (such as NOI18N) (property legalComment, rfe 1385344).
* *
* * Revision 1.5 2005/11/06 11:54:12 oburn

* * Incorporate excellent patch [1344344] Consolidation of regexp checks.
* *
* * Revision 1.3.8.1 2005/10/11 14:26:32 someone
* * Fix for bug 251. The broken bit is fixed
* */
*
* package com.acme.tools;
*
* import com.acme.thing1;
* import com.acme.thing2;
* import com.acme.thing3;
*
* /**
* *
* * <P>
* *   <I>This software is the confidential and proprietary information of
* *   ACME (<B>"Confidential Information"</B>). You shall not
* *   disclose such Confidential Information and shall use it only in
* *   accordance with the terms of the license agreement you entered into
* *   with ACME.</I>
* * </P>
* *
* * © copyright 2002 ACME
* *
* * @author   Some Body
* */
* public class PID extends StateMachine implements WebObject.Constants {
*
* /** javadoc. */
* public static final int A_SETPOINT = 1;
* .
* .
* .
* } // class PID
* </pre>
*
* This checks for the presence of the header, the first 16 lines.
*
*
* Note the following:
*
*
*
* Line 2 and 13 contain the file name. These are checked to make sure they
* are the same, and that they match the class name.
*
*
* The date can be any 4 digit number.

*
*
* <pre>
* <module name="Regexp">
*   <property
*   name="format"
*   value="\A/*\n * (\w*)\.java\n *\n * Copyright \(c\)
*   \d\d\d\d ACME\n * 123 Some St\.\n * Somewhere\.\n *\n
*   * This software is the confidential and proprietary information
*   of ACME\.\n * \("Confidential Information"\)\. You
*   shall not disclose such\n * Confidential Information and shall
*   use it only in accordance with\n * the terms of the license
*   agreement you entered into with ACME\.\n *\n
*   * \$Log: config_misc\.xml,v $
*   * Revision 1\.7 2007/01/16 12:16:35 oburn
*   * Removing all reference to mailing lists
*   * \
*   * Revision 1.6 2005/12/25 16:13:10 o_sukhodolsky
*   * Fix for rfe 1248106 \(TYPECAST is now accepted by NoWhitespaceAfter\)
*   * \
*   * Fix for rfe 953266 \(thanks to Paul Guyot \(pguyot\) for submitting patch\)
*   * IllegalType can be configured to accept some abstract classes which
*   * matches to regexp of illegal type names \(property legalAbstractClassNames\)
*   *
*   * TrailingComment now can be configured to accept some trailing comments
*   * \(such as NOI18N\) \(property legalComment, rfe 1385344\).
*   *
*   * Revision 1.5 2005/11/06 11:54:12 oburn
*   * Incorporate excellent patch \[1344344 \] Consolidation of regexp checks.
*   * \\n(.*\n)*([\w|\s]*(class | interface)\1)"/>
*   <property name="message" value="Correct header not found"/>
* </module>
* </pre>
*
* This checks for the presence of a copyright notice within the class javadoc, lines 24 to 37.
*
* <pre>
* <module name="Regexp">
*   <property
*   name="format"
*   value="(/**\n)(*.*\n)*(* <P>\n *   <I>
*   This software is the confidential and proprietary information of\n
*   *   ACME \(<B>"Confidential Information"</B>
*   \)\. You shall not\n *   disclose such Confidential Information
*   and shall use it only in\n *   accordance with the terms of the
*   license agreement you entered into\n *   with ACME\.</I>\n
*   * </P>\n *\n * © copyright \d\d\d\d ACME\n
*   *\n * @author .*)(\n\s*.*)*/\n[\w|\s]*(class | interface)"/>

```
*   &lt;property name="message"
*     value="Copyright in class/interface Javadoc"/&gt;
*   &lt;property name="duplicateLimit" value="0"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* <b>Note:</b> To search for things that mean something in XML, like &lt;
* you need to escape them like &amp;lt;. This is required so the XML parser
* does not act on them, but instead passes the correct character to the regexp engine.
* </p>
* <p>
* Parent is {@code com.puppycrawl.tools.checkstyle.TreeWalker}
* </p>
* <p>
* Violation Message Keys:
* </p>
* <ul>
* <li>
* {@code duplicate.regexp}
* </li>
* <li>
* {@code illegal.regexp}
* </li>
* <li>
* {@code required.regexp}
* </li>
* </ul>
*
* @since 4.0
*/
```

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/RegexpCheck.java
No license file was found, but licenses were detected in source scan.

header specification (note that this is not the full Apache license header):
line 5: ^\W*Licensed under the Apache License, Version 2\.0 \(the "License"\);$

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/meta/checks/header/RegexpHeaderCheck.xml
No license file was found, but licenses were detected in source scan.

```
// This library is free software; you can redistribute it and/or
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
```

```
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
// Lesser General Public License for more details.
// You should have received a copy of the GNU Lesser General Public
/**
* <p>
* Filter {@code SuppressionXpathFilter} works as
* <a href="https://checkstyle.org/config_filters.html#SuppressionFilter">SuppressionFilter</a>.
* Additionally, filter processes {@code suppress-xpath} elements,
* which contains xpath-expressions. Xpath-expressions are queries for
* suppressed nodes inside the AST tree.
* </p>
* <p>
* Currently, filter does not support the following checks:
* </p>
* <ul id="SuppressionXpathFilter_IncompatibleChecks">
* <li>
* NoCodeInFile (reason is that AST is not generated for a file not containing code)
* </li>
* <li>
* Regexp (reason is at
* <a href="https://github.com/checkstyle/checkstyle/issues/7759#issuecomment-605525287"> #7759</a>)
* </li>
* <li>
* RegexpSinglelineJava (reason is at
* <a href="https://github.com/checkstyle/checkstyle/issues/7759#issuecomment-605525287"> #7759</a>)
* </li>
* </ul>
* <p>
* Also, the filter does not support suppressions inside javadoc reported by Javadoc checks:
* </p>
* <ul id="SuppressionXpathFilter_JavadocChecks">
* <li>
* AtclauseOrder
* </li>
* <li>
* JavadocBlockTagLocation
* </li>
* <li>
* JavadocMethod
* </li>
* <li>
* JavadocMissingLeadingAsterisk
* </li>
* <li>
* JavadocMissingWhitespaceAfterAsterisk
* </li>
```

*
* JavadocParagraph
*
*
* JavadocStyle
*
*
* JavadocTagContinuationIndentation
*
*
* JavadocType
*
*
* MissingDeprecated
*
*
* NonEmptyAtclauseDescription
*
*
* RequireEmptyLineBeforeBlockTagGroup
*
*
* SingleLineJavadoc
*
*
* SummaryJavadoc
*
*
* WriteTag
*
*
*
* Note, that support for these Checks will be available after resolving issue
* #5770.
*
*
* Currently, filter supports the following xpath axes:
*
*
*
* ancestor
*
*
* ancestor-or-self
*
*
* attribute
*

```
* <li>
* child
* </li>
* <li>
* descendant
* </li>
* <li>
* descendant-or-self
* </li>
* <li>
* following
* </li>
* <li>
* following-sibling
* </li>
* <li>
* parent
* </li>
* <li>
* preceding
* </li>
* <li>
* preceding-sibling
* </li>
* <li>
* self
* </li>
* </ul>
* <p>
* You can use the command line helper tool to generate xpath suppressions based on your
* configuration file and input files. See <a href="https://checkstyle.org/cmdline.html">here</a>
* for more details.
* </p>
* <p>
* The suppression file location is checked in following order:
* </p>
* <ol>
* <li>
* as a filesystem location
* </li>
* <li>
* if no file found, and the location starts with either {@code http://} or {@code https://},
* then it is interpreted as a URL
* </li>
* <li>
* if no file found, then passed to the {@code ClassLoader.getResource()} method.
* </li>
* </ol>
```

```
* <p>
* SuppressionXpathFilter can suppress Checks that have Treewalker as parent module.
* </p>
* <ul>
* <li>
* Property {@code file} - Specify the location of the <em>suppressions XML document</em> file.
* Type is {@code java.lang.String}.
* Default value is {@code null}.
* </li>
* <li>
* Property {@code optional} - Control what to do when the file is not existing.
* If optional is set to false the file must exist, or else it ends with error.
* On the other hand if optional is true and file is not found, the filter accepts all audit events.
* Type is {@code boolean}.
* Default value is {@code false}.
* </li>
* </ul>
* <p>
* For example, the following configuration fragment directs the Checker to use a
* {@code SuppressionXpathFilter} with suppressions file {@code config/suppressions.xml}:
* </p>
* <pre>
* &lt;module name=&quot;SuppressionXpathFilter&quot;&gt;
*   &lt;property name=&quot;file&quot; value=&quot;config/suppressions.xml&quot;/&gt;
*   &lt;property name=&quot;optional&quot; value=&quot;false&quot;/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* A <a href="https://checkstyle.org/dtds/suppressions_1_2_xpath_experimental.dtd"><em>
* suppressions XML document</em></a>
* contains a set of {@code suppress} and {@code suppress-xpath} elements,
* where each {@code suppress-xpath} element can have the following attributes:
* </p>
* <ul>
* <li>
* {@code files} - a <a href="https://checkstyle.org/property_types.html#Pattern">Pattern</a>
* matched against the file name associated with an audit event. It is optional.
* </li>
* <li>
* {@code checks} - a <a href="https://checkstyle.org/property_types.html#Pattern">Pattern</a>
* matched against the name of the check associated with an audit event.
* Optional as long as {@code id} or {@code message} is specified.
* </li>
* <li>
* {@code message} - a <a href="https://checkstyle.org/property_types.html#Pattern">Pattern</a>
* matched against the message of the check associated with an audit event.
* Optional as long as {@code checks} or {@code id} is specified.
* </li>
```

```
* <li>
* {@code id} - a <a href="https://checkstyle.org/property_types.html#String">String</a> matched against
* the ID of the check associated with an audit event.
* Optional as long as {@code checks} or {@code message} is specified.
* </li>
* <li>
* {@code query} - a <a href="https://checkstyle.org/property_types.html#String">String</a> xpath query. It is
optional.
* </li>
* </ul>
* <p>
* Each audit event is checked against each {@code suppress} and {@code suppress-xpath} element.
* It is suppressed if all specified attributes match against the audit event.
* </p>
* <p>
* ATTENTION: filtering by message is dependant on runtime locale.
* If project is running in different languages it is better to avoid filtering by message.
* </p>
* <p>
* The following suppressions XML document directs a {@code SuppressionXpathFilter} to reject
* {@code CyclomaticComplexity} violations for all methods with name <i>sayHelloWorld</i> inside
* <i>FileOne</i> and <i>FileTwo</i> files:
* </p>
* <p>
* Currently, xpath queries support one type of attribute {@code @text}. {@code @text} -
* addresses to the text value of the node. For example: variable name, annotation name,
* text content and etc. Only the following token types support {@code @text} attribute:
* {@code TokenTypes.IDENT}, {@code TokenTypes.STRING_LITERAL}, {@code
TokenTypes.CHAR_LITERAL},
* {@code TokenTypes.NUM_LONG}, {@code TokenTypes.NUM_INT}, {@code
TokenTypes.NUM_DOUBLE},
* {@code TokenTypes.NUM_FLOAT}.
* These token types were selected because only their text values are different
* in content from token type and represent text value from file and can be used
* in xpath queries for more accurate results. Other token types always have constant values.
* </p>
* <pre>
* &lt;?xml version=&quot;1.0&quot;?&gt;
*
* &lt;!DOCTYPE suppressions PUBLIC
* &quot;-//Checkstyle//DTD SuppressionXpathFilter Experimental Configuration 1.2//EN&quot;
* &quot;https://checkstyle.org/dtds/suppressions_1_2_xpath_experimental.dtd&quot;&gt;
*
* &lt;suppressions&gt;
*   &lt;suppress-xpath checks=&quot;CyclomaticComplexity&quot;
*   files=&quot;FileOne.java,FileTwo.java&quot;
*   query=&quot;//METHOD_DEF[./IDENT[@text='sayHelloWorld']]&quot;/&gt;
* &lt;/suppressions&gt;
```

```
* </pre>
* <p>
* Suppress checks for package definitions:
* </p>
* <pre>
* &lt;suppress-xpath checks=".*" query="/PACKAGE_DEF"/&gt;
* </pre>
* <p>
* Suppress checks for parent element of the first variable definition:
* </p>
* <pre>
* &lt;suppress-xpath checks=".*" query="(//VARIABLE_DEF)[1]/.."/&gt;
* </pre>
* <p>
* Suppress checks for elements which are either class definitions, either method definitions.
* </p>
* <pre>
* &lt;suppress-xpath checks=".*" query="//CLASS_DEF | //METHOD_DEF"/&gt;
* </pre>
* <p>
* Suppress checks for certain methods:
* </p>
* <pre>
* &lt;suppress-xpath checks=&quot;.*&quot; query=&quot;//METHOD_DEF[./IDENT[@text='getSomeVar'
*         or @text='setSomeVar']]&quot;/&gt;
* </pre>
* <p>
* Suppress checks for variable <i>testVariable</i> inside <i>testMethod</i>
* method inside <i>TestClass</i> class.
* </p>
* <pre>
* &lt;suppress-xpath checks=&quot;.*&quot; query=&quot;//CLASS_DEF[@text='TestClass']
*         //METHOD_DEF[./IDENT[@text='testMethod']]
*         //VARIABLE_DEF[./IDENT[@text='testVariable']]&quot;/&gt;
* </pre>
* <p>
* In the following sample, violations for {@code LeftCurly} check will be suppressed
* for classes with name <i>Main</i> or for methods with name <i>calculate</i>.
* </p>
* <pre>
* &lt;suppress-xpath checks=&quot;LeftCurly&quot; query=&quot;//CLASS_DEF[./IDENT[@text='Main']]//*
*         | //METHOD_DEF[./IDENT[@text='calculate']]/*&quot;/&gt;
* </pre>
* <p>
* The following example demonstrates how to suppress {@code RequireThis} violations
* for variable <i>age</i> inside <i>changeAge</i> method.
* </p>
* <pre>
```

```
* &lt;suppress-xpath checks=&quot;RequireThis&quot;
*     query=&quot;//CLASS_DEF[./IDENT[@text='InputTest']]
*         //METHOD_DEF[./IDENT[@text='changeAge']]//ASSIGN/IDENT[@text='age']&quot;/&gt;
* </pre>
* <pre>
* public class InputTest {
*   private int age = 23;
*
*   public void changeAge() {
*     age = 24; //violation will be suppressed
*   }
* }
* </pre>
* <p>
* Suppress {@code IllegalThrows} violations only for methods with name <i>throwsMethod</i>
* and only for {@code RuntimeException} exceptions. Double colon is used for axis iterations.
* In the following example {@code ancestor} axis is used to iterate all ancestor nodes
* of the current node with type {@code METHOD_DEF} and name <i>throwsMethod</i>.
* Please read more about xpath axes at <a href="https://www.w3schools.com/xml/xpath_axes.asp">
* W3Schools Xpath Axes</a>.
* </p>
* <pre>
* &lt;suppress-xpath checks="IllegalThrows" query="//LITERAL_THROWS
*         /IDENT[@text='RuntimeException' and
*         ./ancestor::METHOD_DEF[./IDENT[@text='throwsMethod']]]"/&gt;
* </pre>
* <pre>
* public class InputTest {
*   public void throwsMethod() throws RuntimeException { // violation will be suppressed
*   }
*
*   public void sampleMethod() throws RuntimeException { // will throw violation here
*   }
* }
* </pre>
* <p>
* The following sample demonstrates how to suppress all violations for method
* itself and all descendants. {@code descendant-or-self} axis iterates through
* current node and all children nodes at any level. Keyword {@code node()}
* selects node elements. Please read more about xpath syntax at
* <a href="https://www.w3schools.com/xml/xpath_syntax.asp">W3Schools Xpath Syntax</a>.
* </p>
* <pre>
* &lt;suppress-xpath checks=".*" query="//METHOD_DEF[./IDENT[@text='legacyMethod']]
*         /descendant-or-self::node()"/&gt;
* </pre>
* <p>
* Some elements can be suppressed in different ways. For example, to suppress
```

```
* violation on variable {@code wordCount} in following code:
* </p>
* <pre>
* public class InputTest {
*     private int wordCount = 11;
* }
* </pre>
* <p>
* You need to look at AST of such code by our CLI tool:
* </p>
* <pre>
* $ java -jar checkstyle-X.XX-all.jar -t InputTest.java
* CLASS_DEF -&gt; CLASS_DEF [1:0]
* |--MODIFIERS -&gt; MODIFIERS [1:0]
* |   `--LITERAL_PUBLIC -&gt; public [1:0]
* |--LITERAL_CLASS -&gt; class [1:7]
* |--IDENT -&gt; InputTest [1:13]
* `--OBJBLOCK -&gt; OBJBLOCK [1:23]
* |--LCURLY -&gt; { [1:23]
* |--VARIABLE_DEF -&gt; VARIABLE_DEF [2:4]
* |   |--MODIFIERS -&gt; MODIFIERS [2:4]
* |   |   `--LITERAL_PRIVATE -&gt; private [2:4]
* |   |--TYPE -&gt; TYPE [2:12]
* |   |   `--LITERAL_INT -&gt; int [2:12]
* |   |--IDENT -&gt; wordCount [2:16]
* |   |--ASSIGN -&gt; = [2:26]
* |   |   `--EXPR -&gt; EXPR [2:28]
* |   |       `--NUM_INT -&gt; 11 [2:28]
* |   `--SEMI -&gt; ; [2:30]
* `--RCURLY -&gt; } [3:0]
* </pre>
* <p>
* The easiest way is to suppress by variable name. As you can see {@code VARIABLE_DEF}
* node refers to variable declaration statement and has child node with token type
* {@code IDENT} which is used for storing class, method, variable names.
* </p>
* <p>
* The following example demonstrates how variable can be queried by its name:
* </p>
* <pre>
* &lt;suppress-xpath checks="." query="//VARIABLE_DEF[
*           ./IDENT[@text='wordCount']]"/&gt;
* </pre>
* <p>
* Another way is to suppress by variable value. Again, if you look at the printed
* AST tree above, you will notice that one of the grandchildren of {@code VARIABLE_DEF}
* node is responsible for storing variable value -{@code NUM_INT} with value <b>11</b>.
* </p>
```

```
* <p>
* The following example demonstrates how variable can be queried by its value,
* same approach applies to {@code String, char, float, double, int, long} data types:
* </p>
* <pre>
* &lt;suppress-xpath checks="." query="//VARIABLE_DEF[.//NUM_INT[@text=11]]"/&gt;
* </pre>
* <p>
* Next example is about suppressing method with certain annotation by its name and element value.
* </p>
* <pre>
* public class InputTest {
*           &#64;Generated("first") // should not be suppressed
*           public void test1() {
*           }
*
*           &#64;Generated("second") // should be suppressed
*           public void test2() {
*           }
*       }
* </pre>
* <p>
* First of all we need to look at AST tree printed by our CLI tool:
* </p>
* <pre>
* $ java -jar checkstyle-X.XX-all.jar -t InputTest.java
* CLASS_DEF -&gt; CLASS_DEF [1:0]
* |--MODIFIERS -&gt; MODIFIERS [1:0]
* |   `--LITERAL_PUBLIC -&gt; public [1:0]
* |--LITERAL_CLASS -&gt; class [1:7]
* |--IDENT -&gt; InputTest [1:13]
* `--OBJBLOCK -&gt; OBJBLOCK [1:23]
* |--LCURLY -&gt; { [1:23]
* |--METHOD_DEF -&gt; METHOD_DEF [2:4]
* |   |--MODIFIERS -&gt; MODIFIERS [2:4]
* |   |   |--ANNOTATION -&gt; ANNOTATION [2:4]
* |   |   |   |--AT -&gt; @ [2:4]
* |   |   |   |--IDENT -&gt; Generated [2:5]
* |   |   |   |--LPAREN -&gt; ( [2:14]
* |   |   |   |--EXPR -&gt; EXPR [2:15]
* |   |   |   |   `--STRING_LITERAL -&gt; "first" [2:15]
* |   |   |   `--RPAREN -&gt; ) [2:22]
* |   |   `--LITERAL_PUBLIC -&gt; public [3:4]
* |   |--TYPE -&gt; TYPE [3:11]
* |   |   `--LITERAL_VOID -&gt; void [3:11]
* |   |--IDENT -&gt; test1 [3:16]
* |   |--LPAREN -&gt; ( [3:21]
* |   |--PARAMETERS -&gt; PARAMETERS [3:22]
```

```
* |   |--RPAREN -&gt; ) [3:22]
* |   `--SLIST -&gt; { [3:24]
* |       `--RCURLY -&gt; } [4:4]
* |--METHOD_DEF -&gt; METHOD_DEF [6:4]
* |   |--MODIFIERS -&gt; MODIFIERS [6:4]
* |   |   |--ANNOTATION -&gt; ANNOTATION [6:4]
* |   |   |   |--AT -&gt; @ [6:4]
* |   |   |   |--IDENT -&gt; Generated [6:5]
* |   |   |   |--LPAREN -&gt; ( [6:14]
* |   |   |   |--EXPR -&gt; EXPR [6:15]
* |   |   |   |   `--STRING_LITERAL -&gt; "second" [6:15]
* |   |   |   `--RPAREN -&gt; ) [6:23]
* |   |   `--LITERAL_PUBLIC -&gt; public [7:4]
* |   |--TYPE -&gt; TYPE [7:11]
* |   |   `--LITERAL_VOID -&gt; void [7:11]
* |   |--IDENT -&gt; test2 [7:16]
* |   |--LPAREN -&gt; ( [7:21]
* |   |--PARAMETERS -&gt; PARAMETERS [7:22]
* |   |--RPAREN -&gt; ) [7:22]
* |   `--SLIST -&gt; { [7:24]
* |       `--RCURLY -&gt; } [8:4]
* `--RCURLY -&gt; } [9:0]
* </pre>
* <p>
* AST node {@code ANNOTATION -> ANNOTATION [6:4]} has direct child
* {@code IDENT -> Generated [6:5]}, therefore can be queried by {@code IDENT} value:
* </p>
* <pre>
* &lt;suppress-xpath checks="." query="//METHOD_DEF[
*           .//ANNOTATION/IDENT[@text='Generated']]"/&gt;
* </pre>
* <p>
* The problem with query above that it will suppress violations for all methods
* with annotation {@code @Generated}. In order to suppress methods with
* {@code @Generated("second")} annotations only, you need to look at AST tree again.
* Value of the {@code ANNOTATION} node is stored inside sub-node with token type
* {@code STRING_LITERAL}. Use the following query to suppress methods with
* {@code @Generated("second")} annotation:
* </p>
* <pre>
* &lt;suppress-xpath checks="." query="//METHOD_DEF[.//ANNOTATION[
*           ./IDENT[@text='Generated'] and ./EXPR/STRING_LITERAL[@text='second']]]"/&gt;
* </pre>
* <p>
* Parent is {@code com.puppycrawl.tools.checkstyle.TreeWalker}
* </p>
*
* @since 8.6
```

* @noinspection NonFinalFieldReferenceInEquals, NonFinalFieldReferencedInHashCode
*/

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SuppressionXpathFilter.java
No license file was found, but licenses were detected in source scan.

// This library is free software; you can redistribute it and/or
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
// Lesser General Public License for more details.
// You should have received a copy of the GNU Lesser General Public
/**
   * Represents a policy for the location of content starting from
   * the same line as {@code /**}.
   * Example:
   * <pre>
   * /** Summary text.
   *   * More details.
   *   */
   * public void method();
   * </pre>
   * This style is also known as "scala" style.
   */
/**
   * Represents a policy for the location of content starting from
   * the next line after {@code /**}.
   * Example:
   * <pre>
   * /**
   * * Summary text.
   * * More details.
   * */
   * public void method();
   * </pre>
   * This style is common to java projects.
   */

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocContentLocationOption.java
No license file was found, but licenses were detected in source scan.

* More details.
* More details.

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/meta/checks/javadoc/JavadocContentLocationCheck.xml
No license file was found, but licenses were detected in source scan.

```
// This library is free software; you can redistribute it and/or
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
// Lesser General Public License for more details.
// You should have received a copy of the GNU Lesser General Public
/**
* <p>
* Checks that a source file begins with a specified header.
* Property {@code headerFile} specifies a file that contains the required header.
* Alternatively, the header specification can be set directly in the
* {@code header} property without the need for an external file.
* </p>
* <p>
* Property {@code ignoreLines} specifies the line numbers to ignore when matching
* lines in a header file. This property is very useful for supporting headers
* that contain copyright dates. For example, consider the following header:
* </p>
* <pre>
* line 1: ////////////////////////////////////////////////////////////////////////////
* line 2: // checkstyle:
* line 3: // Checks Java source code for adherence to a set of rules.
* line 4: // Copyright (C) 2002  Oliver Burn
* line 5: ////////////////////////////////////////////////////////////////////////////
* </pre>
* <p>
* Since the year information will change over time, you can tell Checkstyle
* to ignore line 4 by setting property {@code ignoreLines} to {@code 4}.
* </p>
* <p>
* In default configuration, if header is not specified, the default value
* of header is set to {@code null} and the check does not rise any violations.
* </p>
* <ul>
* <li>
* Property {@code headerFile} - Specify the name of the file containing the required header.
* Type is {@code java.net.URI}.
```

```
* Default value is {@code null}.
* </li>
* <li>
* Property {@code charset} - Specify the character encoding to use when reading the headerFile.
* Type is {@code java.lang.String}.
* Default value is {@code the charset property of the parent
* <a href="https://checkstyle.org/config.html#Checker">Checker</a> module}.
* </li>
* <li>
* Property {@code header} - Specify the required header specified inline.
* Individual header lines must be separated by the string {@code "\n"}
* (even on platforms with a different line separator), see examples below.
* Type is {@code java.lang.String}.
* Default value is {@code null}.
* </li>
* <li>
* Property {@code ignoreLines} - Specify the line numbers to ignore.
* Type is {@code int[]}.
* Default value is {@code ""}.
* </li>
* <li>
* Property {@code fileExtensions} - Specify the file type extension of files to process.
* Type is {@code java.lang.String[]}.
* Default value is {@code ""}.
* </li>
* </ul>
* <p>
* To configure the check such that no violations arise.
* Default values of properties are used.
* </p>
* <pre>
* &lt;module name="Header"/&gt;
* </pre>
* <p>
* To configure the check to use header file {@code "config/java.header"}
* and ignore lines {@code 2}, {@code 3}, and {@code 4} and only process Java files:
* </p>
* <pre>
* &lt;module name="Header"&gt;
*   &lt;property name="headerFile" value="config/java.header"/&gt;
*   &lt;property name="ignoreLines" value="2, 3, 4"/&gt;
*   &lt;property name="fileExtensions" value="java"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* To configure the check to verify that each file starts with the header
* </p>
* <pre>
```

* // Copyright (C) 2004 MyCompany
* // All rights reserved
* </pre>
*
* without the need for an external header file:
*
* <pre>
* <module name="Header">
*   <property name="header"
*   value="// Copyright (C) 2004 MyCompany\n// All rights reserved"/>
* </module>
* </pre>
*
* Parent is {@code com.puppycrawl.tools.checkstyle.Checker}
*
*
* Violation Message Keys:
*
*
*
* {@code header.mismatch}
*
*
* {@code header.missing}
*
*
*
* @since 6.9
*/

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/header/HeaderCheck.java
No license file was found, but licenses were detected in source scan.

//
// checkstyle: Checks Java source code for adherence to a set of rules.
// Copyright (C) 2001-2021 the original author or authors.
//
// This library is free software; you can redistribute it and/or
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
//
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
// Lesser General Public License for more details.

```
//
// You should have received a copy of the GNU Lesser General Public
// License along with this library; if not, write to the Free Software
// Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA  02111-1307  USA
////////////////////////////////////////////////////////////////////////////////

lexer grammar JavaLanguageLexer;

channels { COMMENTS }
//Please add new tokens only in the end of list! Otherwise you break compatibility!
tokens {
  // Root of Checkstyle AST
  COMPILATION_UNIT,

  // Maintain compatibility in token numbering
  PLACEHOLDER1, NULL_TREE_LOOKAHEAD,

  // Pre-1.4 tokens
  BLOCK, MODIFIERS, OBJBLOCK, SLIST, CTOR_DEF, METHOD_DEF, VARIABLE_DEF,
  INSTANCE_INIT, STATIC_INIT, TYPE, CLASS_DEF, INTERFACE_DEF,
  PACKAGE_DEF, ARRAY_DECLARATOR, EXTENDS_CLAUSE, IMPLEMENTS_CLAUSE,
  PARAMETERS, PARAMETER_DEF, LABELED_STAT, TYPECAST, INDEX_OP,
  POST_INC, POST_DEC, METHOD_CALL, EXPR, ARRAY_INIT,
  IMPORT, UNARY_MINUS, UNARY_PLUS, CASE_GROUP, ELIST, FOR_INIT, FOR_CONDITION,
  FOR_ITERATOR, EMPTY_STAT, FINAL, ABSTRACT,
  STRICTFP, SUPER_CTOR_CALL, CTOR_CALL,

  // ANTLR-generated pre-1.4 tokens now listed here to preserve their numerical
  // order so as to make all future version of this grammar backwardly compatible
  LITERAL_PACKAGE,SEMI,LITERAL_IMPORT,LBRACK,RBRACK,
  LITERAL_VOID,LITERAL_BOOLEAN,LITERAL_BYTE,
  LITERAL_CHAR,LITERAL_SHORT,LITERAL_INT,
  LITERAL_FLOAT,LITERAL_LONG,LITERAL_DOUBLE,
  IDENT,DOT,STAR,LITERAL_PRIVATE,LITERAL_PUBLIC,
  LITERAL_PROTECTED,LITERAL_STATIC,
  LITERAL_TRANSIENT,LITERAL_NATIVE,
  LITERAL_SYNCHRONIZED,LITERAL_VOLATILE,

  // Please add new tokens only in the end of list! Otherwise you break compatibility!
  LITERAL_CLASS,LITERAL_EXTENDS,
  LITERAL_INTERFACE,LCURLY,RCURLY,COMMA,
  LITERAL_IMPLEMENTS,LPAREN,RPAREN,LITERAL_THIS,
  LITERAL_SUPER,ASSIGN,LITERAL_THROWS,COLON,
  LITERAL_IF,LITERAL_WHILE,LITERAL_DO,
  LITERAL_BREAK,LITERAL_CONTINUE,LITERAL_RETURN,
  LITERAL_SWITCH,LITERAL_THROW,LITERAL_FOR,
  LITERAL_ELSE,LITERAL_CASE,LITERAL_DEFAULT,
```

```
  // Please add new tokens only in the end of list! Otherwise you break compatibility!
  LITERAL_TRY,LITERAL_CATCH,LITERAL_FINALLY,
  PLUS_ASSIGN,MINUS_ASSIGN,STAR_ASSIGN,DIV_ASSIGN,MOD_ASSIGN,SR_ASSIGN,
  BSR_ASSIGN,SL_ASSIGN,BAND_ASSIGN,BXOR_ASSIGN,BOR_ASSIGN,QUESTION,
  LOR,LAND,BOR,BXOR,BAND,NOT_EQUAL,EQUAL,LT,GT,LE,GE,
  LITERAL_INSTANCEOF,SL,SR,BSR,PLUS,MINUS,DIV,MOD,
  INC,DEC,BNOT,LNOT,LITERAL_TRUE,LITERAL_FALSE,
  LITERAL_NULL,LITERAL_NEW,NUM_INT,CHAR_LITERAL,
  STRING_LITERAL,NUM_FLOAT,NUM_LONG,NUM_DOUBLE,WS,SINGLE_LINE_COMMENT,
  BLOCK_COMMENT_BEGIN,ESC,HEX_DIGIT,VOCAB,EXPONENT,FLOAT_SUFFIX,

  // Please add new tokens only in the end of list! Otherwise you break compatibility!
  // Token for Java 1.4 language enhancements
  ASSERT,

  // Tokens for Java 1.5 language enhancements
  STATIC_IMPORT, ENUM, ENUM_DEF, ENUM_CONSTANT_DEF, FOR_EACH_CLAUSE,
  ANNOTATION_DEF, ANNOTATIONS, ANNOTATION, ANNOTATION_MEMBER_VALUE_PAIR,
ANNOTATION_FIELD_DEF,
  ANNOTATION_ARRAY_INIT, TYPE_ARGUMENTS, TYPE_ARGUMENT, TYPE_PARAMETERS,
  TYPE_PARAMETER, WILDCARD_TYPE, TYPE_UPPER_BOUNDS, TYPE_LOWER_BOUNDS, AT,
ELLIPSIS,
  GENERIC_START, GENERIC_END, TYPE_EXTENSION_AND,

  // Please add new tokens only in the end of list! Otherwise you break compatibility!

  // token which was not included to grammar initially
  // we need to put it to the end to maintain binary compatibility
  // with previous versions
  DO_WHILE,

  // Tokens for Java 1.7 language enhancements
  RESOURCE_SPECIFICATION, RESOURCES, RESOURCE,

  // Tokens for Java 1.8
  DOUBLE_COLON,  METHOD_REF, LAMBDA,

  // Support of java comments has been extended
  BLOCK_COMMENT_END,COMMENT_CONTENT,

  // Need to add these here to preserve order of tokens
  SINGLE_LINE_COMMENT_CONTENT, BLOCK_COMMENT_CONTENT, STD_ESC,
  BINARY_DIGIT, ID_START, ID_PART, INT_LITERAL, LONG_LITERAL,
  FLOAT_LITERAL, DOUBLE_LITERAL, HEX_FLOAT_LITERAL, HEX_DOUBLE_LITERAL,
  SIGNED_INTEGER, BINARY_EXPONENT,

  PATTERN_VARIABLE_DEF, RECORD_DEF, LITERAL_RECORD,
  RECORD_COMPONENTS, RECORD_COMPONENT_DEF, COMPACT_CTOR_DEF,
```

```
  TEXT_BLOCK_LITERAL_BEGIN, TEXT_BLOCK_CONTENT, TEXT_BLOCK_LITERAL_END,
  LITERAL_YIELD, SWITCH_RULE,

  LITERAL_NON_SEALED, LITERAL_SEALED, LITERAL_PERMITS,
  PERMITS_CLAUSE
}

@header {
import com.puppycrawl.tools.checkstyle.grammar.CommentListener;
import com.puppycrawl.tools.checkstyle.grammar.CrAwareLexerSimulator;
}

@lexer::members {
  /**
   * We need to create a different constructor in order to use our
   * own implementation of the LexerATNSimulator. This is the
   * reason for the unused 'crAwareConstructor' argument.
   *
   * @param input the character stream to tokenize
   * @param crAwareConstructor dummy parameter
   */
  public JavaLanguageLexer(CharStream input, boolean crAwareConstructor) {
    super(input);
    _interp = new CrAwareLexerSimulator(this, _ATN, _decisionToDFA, _sharedContextCache);
  }

  private CommentListener commentListener = null;

  /**
   * Sets the CommentListener for the lexer.
   *
   * @param commentListener the commentListener to use in this lexer
   */
  public void setCommentListener(CommentListener commentListener){
        this.commentListener = commentListener;
  }

  /** Tracks the starting line of a block comment. */
  int startLine = -1;

  /** Tracks the starting column of a block comment. */
  int startCol = -1;
}

// Keywords and restricted identifiers
ABSTRACT:               'abstract';
ASSERT:                 'assert';
LITERAL_BOOLEAN:        'boolean';
```

```
LITERAL_BREAK:          'break';
LITERAL_BYTE:           'byte';
LITERAL_CASE:           'case';
LITERAL_CATCH:          'catch';
LITERAL_CHAR:           'char';
LITERAL_CLASS:          'class';
LITERAL_CONST:          'const';
LITERAL_CONTINUE:       'continue';
LITERAL_DEFAULT:        'default';
LITERAL_DO:             'do';
LITERAL_DOUBLE:         'double';
LITERAL_ELSE:           'else';
ENUM:                   'enum';
EXTENDS_CLAUSE:         'extends';
FINAL:                  'final';
LITERAL_FINALLY:        'finally';
LITERAL_FLOAT:          'float';
LITERAL_FOR:            'for';
LITERAL_IF:             'if';
LITERAL_GOTO:           'goto';
LITERAL_IMPLEMENTS:     'implements';
IMPORT:                 'import';
LITERAL_INSTANCEOF:     'instanceof';
LITERAL_INT:            'int';
LITERAL_INTERFACE:      'interface';
LITERAL_LONG:           'long';
LITERAL_NATIVE:         'native';
LITERAL_NEW:            'new';
LITERAL_PACKAGE:        'package';
LITERAL_PRIVATE:        'private';
LITERAL_PROTECTED:      'protected';
LITERAL_PUBLIC:         'public';
LITERAL_RETURN:         'return';
LITERAL_SHORT:          'short';
LITERAL_STATIC:         'static';
STRICTFP:               'strictfp';
LITERAL_SUPER:          'super';
LITERAL_SWITCH:         'switch';
LITERAL_SYNCHRONIZED:   'synchronized';
LITERAL_THIS:           'this';
LITERAL_THROW:          'throw';
LITERAL_THROWS:         'throws';
LITERAL_TRANSIENT:      'transient';
LITERAL_TRY:            'try';
LITERAL_VOID:           'void';
LITERAL_VOLATILE:       'volatile';
LITERAL_WHILE:          'while';
LITERAL_RECORD:         'record';
```

```
LITERAL_YIELD:          'yield';
LITERAL_NON_SEALED:     'non-sealed';
LITERAL_SEALED:         'sealed';
LITERAL_PERMITS:        'permits';

// Literals
DECIMAL_LITERAL_LONG:   ('0' | [1-9] (Digits? | '_'+ Digits)) [lL];
DECIMAL_LITERAL:        ('0' | [1-9] (Digits? | '_'+ Digits));

HEX_LITERAL_LONG:       '0' [xX] [0-9a-fA-F] ([0-9a-fA-F_]* [0-9a-fA-F])? [lL];
HEX_LITERAL:            '0' [xX] [0-9a-fA-F] ([0-9a-fA-F_]* [0-9a-fA-F])?;

OCT_LITERAL_LONG:       '0' '_'* [0-7] ([0-7_]* [0-7])? [lL];
OCT_LITERAL:            '0' '_'* [0-7] ([0-7_]* [0-7])?;

BINARY_LITERAL_LONG:    '0' [bB] [01] ([01_]* [01])? [lL];
BINARY_LITERAL:         '0' [bB] [01] ([01_]* [01])?;


DOUBLE_LITERAL:         (Digits '.' Digits? | '.' Digits) ExponentPart? [dD]
        |       Digits (ExponentPart [dD] | [dD])
        ;
FLOAT_LITERAL:          (Digits '.' Digits? | '.' Digits) ExponentPart? [fF]?
        |       Digits (ExponentPart [fF]? | [fF])
        ;


HEX_DOUBLE_LITERAL:     '0' [xX] (HexDigits '.'? | HexDigits? '.' HexDigits)
                [pP] [+-]? Digits [dD];

HEX_FLOAT_LITERAL:      '0' [xX] (HexDigits '.'? | HexDigits? '.' HexDigits)
                [pP] [+-]? Digits [fFdD]?;

LITERAL_TRUE:           'true';

LITERAL_FALSE:          'false';

CHAR_LITERAL:           '\'' (EscapeSequence | ~['\\\r\n]) '\'';

STRING_LITERAL:         '"' (EscapeSequence | ~["\\\r\n])* '"';

TEXT_BLOCK_LITERAL_BEGIN: '"' '"' '"' -> pushMode(TextBlock);

LITERAL_NULL:           'null';

// Separators

LPAREN:                 '(';
```

```
RPAREN:          ')';
LCURLY:          '{';
RCURLY:          '}';
LBRACK:          '[';
RBRACK:          ']';
SEMI:          ';';
COMMA:          ',';
DOT:          '.';

// Operators

ASSIGN:          '=';
GT:          '>';
LT:          '<';
LNOT:          '!';
BNOT:          '~';
QUESTION:          '?';
COLON:          ':';
EQUAL:          '==';
LE:          '<=';
GE:          '>=';
NOT_EQUAL:          '!=';
LAND:          '&&';
LOR:          '||';
INC:          '++';
DEC:          '--';
PLUS:          '+';
MINUS:          '-';
STAR:          '*';
DIV:          '/';
BAND:          '&';
BOR:          '|';
BXOR:          '^';
MOD:          '%';
//SR:          '>>'; handled in parser
//SL:          '<<'; handled in parser

PLUS_ASSIGN:          '+=';
MINUS_ASSIGN:          '-=';
STAR_ASSIGN:          '*=';
DIV_ASSIGN:          '/=';
BAND_ASSIGN:          '&=';
BOR_ASSIGN:          '|=';
BXOR_ASSIGN:          '^=';
MOD_ASSIGN:          '%=';
SL_ASSIGN:          '<<=';
SR_ASSIGN:          '>>=';
//BSR:          '>>>'; handled in parser
```

```
BSR_ASSIGN:             '>>>=';

// Java 8 tokens

LAMBDA:                 '->';
DOUBLE_COLON:           '::';

// Additional symbols not defined in the lexical specification

AT:                     '@';
ELLIPSIS:               '...';

// Whitespace and comments

WS:                     [ \t\r\n\u000C]+ -> skip;

BLOCK_COMMENT_BEGIN:
   // Match block comment start delimiter, set start position
   '/*' { startLine = _tokenStartLine; startCol = _tokenStartCharPositionInLine; }

    // Match content of comment, non-greedy
   .*?

   // Match comment ending delimiter
   '*/'
   {
      // Trim delimiters from comment text
      setText(getText().substring(2, getText().length() - 2));
      // Report comment
      commentListener.reportBlockComment("/*", startLine, startCol,
          _interp.getLine(), _interp.getCharPositionInLine() - 1);
      startLine = startCol = -1;
   } -> channel(COMMENTS);

SINGLE_LINE_COMMENT:
   // Match single line comment delimiter
   '//'
   {
      // Report comment
      commentListener.reportSingleLineComment("//",
          _tokenStartLine, _tokenStartCharPositionInLine);
   }
    // Match content, including terminating newline (or nothing if EOF)
   ~[\r\n]* ( '\n' | '\r' ('\n')? | /** nothing */ )
   {
      // Trim delimiters from comment text
      setText(getText().substring(2));
   } -> channel(COMMENTS);
```

```
// Identifiers

IDENT:        Letter LetterOrDigit*;

// Fragment rules

fragment ExponentPart
  : [eE] [+-]? Digits
  ;

fragment EscapeSequence
  : '\\'
    ('u'+ ( '0' '0' '5' ('c' | 'C' )
          ('\\' 'u'+ HexDigit HexDigit HexDigit HexDigit | StandardEscape)
        | HexDigit HexDigit HexDigit HexDigit
        )
    | StandardEscape
    )
  ;

fragment StandardEscape
  : [btnfrs'"\\]
  | [0-3] (([0-7]) [0-7]?)?
  | [4-7] ([0-9])?
  ;

fragment HexDigits
  : HexDigit ((HexDigit | '_')* HexDigit)?
  ;

fragment HexDigit
  : [0-9a-fA-F]
  ;

fragment Digits
  : [0-9] ([0-9_]* [0-9])?
  ;

fragment LetterOrDigit
  : Letter
  | [0-9]
  ;

fragment Letter
  // these are the "java letters" below 0x7F
  : [a-zA-Z$_]
  // covers all characters above 0x7F which are not a surrogate
```

```
  | ~[\u0000-\u007F\uD800-\uDBFF]
  // covers UTF-16 surrogate pairs encodings for U+10000 to U+10FFFF
  | [\uD800-\uDBFF] [\uDC00-\uDFFF]
  ;

// Text block lexical mode
mode TextBlock;
  TEXT_BLOCK_CONTENT
    : ( TwoDoubleQuotes
      | OneDoubleQuote
      | Newline
      | ~""
      | TextBlockStandardEscape
      )+
    ;

  TEXT_BLOCK_LITERAL_END
    : "" "" "" -> popMode
    ;

  // Text block fragment rules
  fragment TextBlockStandardEscape
    :   '\\' [btnfrs'"\\]
    ;

  fragment Newline
    :  '\n' | '\r' ('\n')?
    ;

  fragment TwoDoubleQuotes
    :   """" ( Newline | ~"" )
    ;

  fragment OneDoubleQuote
    :   "" ( Newline | ~"" )
    ;

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-
jar/com/puppycrawl/tools/checkstyle/grammar/java/JavaLanguageLexer.g4
No license file was found, but licenses were detected in source scan.

for more details.

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-
jar/com/puppycrawl/tools/checkstyle/meta/filters/SuppressionXpathFilter.xml
```

No license file was found, but licenses were detected in source scan.

```
// This library is free software; you can redistribute it and/or
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
// Lesser General Public License for more details.
// You should have received a copy of the GNU Lesser General Public
/**
* <p>
* Checks that the Javadoc content begins from the same position
* for all Javadoc comments in the project. Any leading asterisks and spaces
* are not counted as the beginning of the content and are therefore ignored.
* </p>
* <p>
* It is possible to enforce two different styles:
* </p>
* <ul>
* <li>
* {@code first_line} - Javadoc content starts from the first line:
* <pre>
* &#47;** Summary text.
*   * More details.
*   *&#47;
* public void method();
* </pre>
* </li>
* <li>
* {@code second_line} - Javadoc content starts from the second line:
* <pre>
* &#47;**
*   * Summary text.
*   * More details.
*   *&#47;
* public void method();
* </pre>
* </li>
* </ul>
* <p>
* This check does not validate the Javadoc summary itself nor its presence.
* The check will not report any violations for missing or malformed javadoc summary.
* To validate the Javadoc summary use
* <a href="https://checkstyle.org/config_javadoc.html#SummaryJavadoc">SummaryJavadoc</a> check.
* </p>
* <p>
```

```
* The <a href="https://docs.oracle.com/en/java/javase/11/docs/specs/doc-comment-spec.html">
* Documentation Comment Specification</a> permits leading asterisks on the first line.
* For these Javadoc comments:
* </p>
* <pre>
* &#47;***
*   * Some text.
*   *&#47;
* &#47;*************
*   * Some text.
*   *&#47;
* &#47;**          **
*   * Some text.
*   *&#47;
* </pre>
* <p>
* The documentation generated will be just "Some text." without any asterisks.
* Since these asterisks will not appear in the generated documentation,
* they should not be considered as the beginning of the Javadoc content.
* In such cases, the check assumes that the Javadoc content begins on the second line.
* </p>
* <ul>
* <li>
* Property {@code location} - Specify the policy on placement of the Javadoc content.
* Type is {@code com.puppycrawl.tools.checkstyle.checks.javadoc.JavadocContentLocationOption}.
* Default value is {@code second_line}.
* </li>
* </ul>
* <p>
* To configure the default check to validate that the Javadoc content starts from the second line:
* </p>
* <pre>
* &lt;module name="JavadocContentLocationCheck"/&gt;
* </pre>
* <p>
* This setting produces a violation for each multi-line comment starting
* on the same line as the initial asterisks:
* </p>
* <pre>
* &#47;** This comment causes a violation because it starts from the first line
*   * and spans several lines.
*   *&#47;
* &#47;**
*   * This comment is OK because it starts from the second line.
*   *&#47;
* &#47;** This comment is OK because it is on the single line. *&#47;
* </pre>
* <p>
```

```
* To ensure that Javadoc content starts from the first line:
* </p>
* <pre>
* &lt;module name="JavadocContentLocationCheck"&gt;
*   &lt;property name="location" value="first_line"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* This setting produces a violation for each comment not
* starting on the same line as the initial asterisks:
* </p>
* <pre>
* &#47;** This comment is OK because it starts on the first line.
*    * There may be additional text.
*    *&#47;
* &#47;**
*   * This comment causes a violation because it starts on the second line.
*   *&#47;
* &#47;** This single-line comment also is OK. *&#47;
* </pre>
* <p>
* Parent is {@code com.puppycrawl.tools.checkstyle.TreeWalker}
* </p>
* <p>
* Violation Message Keys:
* </p>
* <ul>
* <li>
* {@code javadoc.content.first.line}
* </li>
* <li>
* {@code javadoc.content.second.line}
* </li>
* </ul>
*
* @since 8.27
*/
```

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocContentLocationCheck.java
No license file was found, but licenses were detected in source scan.

```
////////////////////////////////////////////////////////////////////////////////
// checkstyle: Checks Java source code for adherence to a set of rules.
// Copyright (C) 2001-2021 the original author or authors.
//
// This library is free software; you can redistribute it and/or
```

```
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
//
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
// Lesser General Public License for more details.
//
// You should have received a copy of the GNU Lesser General Public
// License along with this library; if not, write to the Free Software
// Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA  02111-1307  USA
////////////////////////////////////////////////////////////////////////////////

parser grammar JavaLanguageParser;

options { tokenVocab=JavaLanguageLexer; }

@parser::members {
  /**
  * This value tracks the depth of a switch expression. Along with the
  * IDENT to id rule at the end of the parser, this value helps us
  * to know if the "yield" we are parsing is an IDENT, method call, class,
  * field, etc. or if it is a java 13+ yield statement. Positive values
  * indicate that we are within a (possibly nested) switch expression.
  */
  private int switchBlockDepth = 0;

  /**
   * Checks if next statement is a yield statement.
   *
   * @return true if next statement is a yield statement.
   */
  private boolean isYieldStatement() {
     return _input.LT(1).getType() == JavaLanguageLexer.LITERAL_YIELD && switchBlockDepth > 0;
  }
}

compilationUnit
  : packageDeclaration? importDeclaration* typeDeclaration* EOF
  ;

packageDeclaration
  : annotations[true] LITERAL_PACKAGE qualifiedName SEMI
  ;

importDeclaration
  : IMPORT LITERAL_STATIC? qualifiedName (DOT STAR)? SEMI    #importDec
```

```
  | SEMI                                  #singleSemiImport
  ;

typeDeclaration
  : mods+=modifier*
    type=types[$ctx.mods]
  | semi+=SEMI+
  ;

types[List<ModifierContext> mods]
  : classDeclaration[mods]
  | enumDeclaration[mods]
  | interfaceDeclaration[mods]
  | annotationTypeDeclaration[mods]
  | recordDeclaration[mods]
  ;


modifier
  : annotation
  | LITERAL_PUBLIC
  | LITERAL_PROTECTED
  | LITERAL_PRIVATE
  | LITERAL_STATIC
  | ABSTRACT
  | LITERAL_DEFAULT
  | FINAL
  | STRICTFP
  | LITERAL_NATIVE
  | LITERAL_SYNCHRONIZED
  | LITERAL_TRANSIENT
  | LITERAL_VOLATILE
  | LITERAL_NON_SEALED
  | LITERAL_SEALED
  ;

variableModifier
  : FINAL
  | annotation
  ;

classDeclaration[List<ModifierContext> mods]
  : LITERAL_CLASS id typeParameters?
    classExtends?
    implementsClause?
    permittedSubclassesAndInterfaces?
    classBody
  ;
```

```
recordDeclaration[List<ModifierContext> mods]
  : LITERAL_RECORD id typeParameters?
    recordComponentsList
    implementsClause?
    recordBody
  ;

recordComponentsList
  : LPAREN recordComponents? RPAREN
  ;

recordComponents
  : recordComponent (COMMA recordComponent)* (COMMA lastRecordComponent)?
  | lastRecordComponent
  ;

recordComponent
  : annotations[true] type=typeType[true]
    id
  ;

lastRecordComponent
  : annotations[true] type=typeType[true]
    ELLIPSIS id
  ;

recordBody
  : LCURLY recordBodyDeclaration* RCURLY
  ;

recordBodyDeclaration
  : compactConstructorDeclaration
  | classBodyDeclaration
  ;

compactConstructorDeclaration
  : mods+=modifier* id constructorBlock
  ;

classExtends
  : EXTENDS_CLAUSE type=typeType[false]
  ;

implementsClause
  : LITERAL_IMPLEMENTS typeList
  ;
```

```
typeParameters
  : LT typeParameter (COMMA typeParameter)* GT
  ;

typeParameter
  : annotations[false] id typeUpperBounds?
  ;

typeUpperBounds
  : EXTENDS_CLAUSE annotations[false] typeBound
  ;

typeBound
  : typeBoundType (BAND typeBoundType)*
  ;

typeBoundType
  : annotations[false] classOrInterfaceOrPrimitiveType arrayDeclarator*
  ;

enumDeclaration[List<ModifierContext> mods]
  : ENUM id implementsClause?
    enumBody
  ;

enumBody
  : LCURLY enumConstants? COMMA? enumBodyDeclarations? RCURLY
  ;

enumConstants
  : enumConstant (COMMA enumConstant)*
  ;

enumConstant
  : annotations[true] id arguments? classBody?
  ;

enumBodyDeclarations
  : SEMI classBodyDeclaration*
  ;

interfaceDeclaration[List<ModifierContext> mods]
  : LITERAL_INTERFACE id typeParameters?
    interfaceExtends?
    permittedSubclassesAndInterfaces?
    interfaceBody
  ;
```

```
interfaceExtends
  : EXTENDS_CLAUSE typeList
  ;

classBody
  : LCURLY classBodyDeclaration* RCURLY
  ;

interfaceBody
  : LCURLY interfaceBodyDeclaration* RCURLY
  ;

classBodyDeclaration
  : SEMI                                          #emptyClass
  | LITERAL_STATIC? block                                 #classBlock
  | mods+=modifier*
    memberDeclaration[{((ClassDefContext) _localctx).mods}]       #classDef
  ;

memberDeclaration[List<ModifierContext> mods]
  : recordDeclaration[mods]
  | methodDeclaration[mods]
  | fieldDeclaration[mods]
  | constructorDeclaration[mods]
  | interfaceDeclaration[mods]
  | annotationTypeDeclaration[mods]
  | classDeclaration[mods]
  | enumDeclaration[mods]
  ;

methodDeclaration[List<ModifierContext> mods]
  : typeParams=typeParameters? type=typeType[true] id
    formalParameters (cStyleArrDec+=arrayDeclarator*)
    throwsList?
    methodBody
  ;

methodBody
  : block
  | SEMI
  ;

throwsList
  : LITERAL_THROWS qualifiedNameList
  ;

constructorDeclaration[List<ModifierContext> mods]
  : typeParameters? id formalParameters
```

```
    throwsList? constructorBody=constructorBlock
  ;

fieldDeclaration[List<ModifierContext> mods]
  : type=typeType[true]
    variableDeclarators[$mods, $ctx.type]
    SEMI
  ;

interfaceBodyDeclaration
  : mods+=modifier* interfaceMemberDeclaration[$ctx.mods]
  | SEMI
  ;

interfaceMemberDeclaration[List<ModifierContext> mods]
  : fieldDeclaration[mods]
  | interfaceMethodDeclaration[mods]
  | interfaceDeclaration[mods]
  | annotationTypeDeclaration[mods]
  | classDeclaration[mods]
  | recordDeclaration[mods]
  | enumDeclaration[mods]
  ;


// Early versions of Java allows brackets after the method name, eg.
// public int[] return2DArray() [] { ... }
// is the same as
// public int[][] return2DArray() { ... }
interfaceMethodDeclaration[List<ModifierContext> mods]
  : typeParameters?
    type=typeType[true]
    id formalParameters
    cStyleArrDec+=arrayDeclarator*
    throwsList? methodBody
  ;

variableDeclarators[List<ModifierContext> mods, TypeTypeContext type]
  : variableDeclarator[mods, type] (COMMA variableDeclarator[mods, type])*
  ;

variableDeclarator[List<ModifierContext> mods, TypeTypeContext type]
  : id arrayDeclarator* (ASSIGN variableInitializer)?
  ;

variableDeclaratorId[List<VariableModifierContext> mods, ParserRuleContext type]
  : (LITERAL_THIS | (qualifiedName (DOT LITERAL_THIS)?)) arrayDeclarator*
  ;
```

```
variableInitializer
  : arrayInitializer
  | expression
  ;

arrayInitializer
  : LCURLY
    (variableInitializer (COMMA variableInitializer)*)? COMMA?
    RCURLY
  ;

classOrInterfaceType[boolean createImaginaryNode]
  : annotations[false] id typeArguments?
    extended+=classOrInterfaceTypeExtended*
  ;

classOrInterfaceTypeExtended
  : DOT annotations[false] id typeArguments?
  ;

typeArgument
  : typeType[false]                                    #simpleTypeArgument
  | annotations[false] QUESTION
      (
        ( upperBound=EXTENDS_CLAUSE | lowerBound=LITERAL_SUPER )
        typeType[false]
      )?                                               #wildCardTypeArgument
  ;

qualifiedNameList
  : annotations[false]
    qualifiedName (COMMA annotations[false] qualifiedName)*
  ;

formalParameters
  : LPAREN formalParameterList? RPAREN
  ;

formalParameterList
  : formalParameter (COMMA formalParameter)* (COMMA lastFormalParameter)?
  | lastFormalParameter
  ;

formalParameter
  : mods+=variableModifier* type=typeType[true]
    variableDeclaratorId[$ctx.mods, $ctx.type]
  ;
```

```
lastFormalParameter
  : mods+=variableModifier* type=typeType[true] annotations[false]
    ELLIPSIS variableDeclaratorId[$ctx.mods, $ctx.type]
  ;

qualifiedName
  : id extended+=qualifiedNameExtended*
  ;

qualifiedNameExtended
  : DOT annotations[false] id
  ;

literal
  : integerLiteral
  | floatLiteral
  | textBlockLiteral
  | CHAR_LITERAL
  | STRING_LITERAL
  | LITERAL_TRUE
  | LITERAL_FALSE
  | LITERAL_NULL
  ;

integerLiteral
  : DECIMAL_LITERAL_LONG
  | DECIMAL_LITERAL
  | HEX_LITERAL_LONG
  | HEX_LITERAL
  | OCT_LITERAL_LONG
  | OCT_LITERAL
  | BINARY_LITERAL_LONG
  | BINARY_LITERAL
  ;

floatLiteral
  : DOUBLE_LITERAL
  | FLOAT_LITERAL
  | HEX_DOUBLE_LITERAL
  | HEX_FLOAT_LITERAL
  ;

textBlockLiteral
  : TEXT_BLOCK_LITERAL_BEGIN TEXT_BLOCK_CONTENT TEXT_BLOCK_LITERAL_END
  ;

annotations[boolean createImaginaryNode]
```

```
  : anno+=annotation*
  ;

annotation
  : AT qualifiedName
    (LPAREN ( elementValuePairs | elementValue )? RPAREN)?
  ;

elementValuePairs
  : elementValuePair (COMMA elementValuePair)*
  ;

elementValuePair
  : id ASSIGN elementValue
  ;

elementValue
  : expression
  | annotation
  | elementValueArrayInitializer
  ;

elementValueArrayInitializer
  : LCURLY (elementValue (COMMA elementValue)*)? (COMMA)? RCURLY
  ;

annotationTypeDeclaration[List<ModifierContext> mods]
  : AT LITERAL_INTERFACE id annotationTypeBody
  ;

annotationTypeBody
  : LCURLY (annotationTypeElementDeclaration)* RCURLY
  ;

annotationTypeElementDeclaration
  : mods+=modifier* annotationTypeElementRest[$ctx.mods]
  | SEMI // this is not allowed by the grammar, but apparently allowed by the actual compiler
  ;

annotationTypeElementRest[List<ModifierContext> mods]
  : type=typeType[true]
    ( annotationMethodRest[mods, ((AnnotationFieldContext) _localctx).type]
    | annotationConstantRest[mods, ((AnnotationFieldContext) _localctx).type]
    )
    SEMI                                        #annotationField
  | type=typeType[true]  SEMI                   #annotationType
  | classDeclaration[mods] SEMI?                #annotationType
  | recordDeclaration[mods] SEMI?               #annotationType
```

```
  | interfaceDeclaration[mods] SEMI?                    #annotationType
  | enumDeclaration[mods] SEMI?                        #annotationType
  | annotationTypeDeclaration[mods] SEMI?                  #annotationType
  ;

annotationMethodRest[List<ModifierContext> mods, TypeTypeContext type]
  : id LPAREN RPAREN cStyleArrDec+=arrayDeclarator* defaultValue?
  ;

annotationConstantRest[List<ModifierContext> mods, TypeTypeContext type]
  : variableDeclarators[mods, type]
  ;

defaultValue
  : LITERAL_DEFAULT elementValue
  ;

// STATEMENTS / BLOCKS
constructorBlock
  : LCURLY
    explicitConstructorInvocation?
    blockStatement*
    RCURLY
  ;

explicitConstructorInvocation
  : typeArguments?
    (LITERAL_THIS | LITERAL_SUPER)
    arguments SEMI                                  #explicitCtorCall
  | expr DOT typeArguments? LITERAL_SUPER arguments SEMI          #primaryCtorCall
  ;

block
  : LCURLY blockStatement* RCURLY
  ;

blockStatement
  : {!isYieldStatement()}? localVariableDeclaration SEMI          #localVar
  | statement                                     #stat
  | localTypeDeclaration                              #localType
  ;

localVariableDeclaration
  : mods+=modifier* type=typeType[true]
    variableDeclarators[$ctx.mods, $ctx.type]
  ;

localTypeDeclaration
```

```
  : mods+=modifier*
    ( classDeclaration[$ctx.mods]
    | enumDeclaration[$ctx.mods]
    | interfaceDeclaration[$ctx.mods]
    | recordDeclaration[$ctx.mods]
    )
  | SEMI
  ;

statement
  : blockLabel=block                                    #blockStat
  | ASSERT expression (COLON expression)? SEMI          #assertExp
  | LITERAL_IF parExpression statement elseStat?        #ifStat
  | LITERAL_FOR forControl statement                    #forStat
  | LITERAL_WHILE parExpression statement               #whileStat
  | LITERAL_DO statement LITERAL_WHILE parExpression SEMI  #doStat
  | LITERAL_TRY block (catchClause+ finallyBlock? | finallyBlock)  #tryStat
  | LITERAL_TRY resourceSpecification block catchClause* finallyBlock?  #tryWithResourceStat
  | LITERAL_YIELD expression SEMI                       #yieldStat
  | switchExpressionOrStatement                         #switchStat
  | LITERAL_SYNCHRONIZED parExpression block            #syncStat
  | LITERAL_RETURN expression? SEMI                     #returnStat
  | LITERAL_THROW expression SEMI                       #throwStat
  | LITERAL_BREAK id? SEMI                              #breakStat
  | LITERAL_CONTINUE id? SEMI                           #continueStat
  | SEMI                                                #emptyStat
  | statementExpression=expression SEMI                 #expStat
  | id COLON statement                                  #labelStat
  ;

switchExpressionOrStatement
  : LITERAL_SWITCH parExpression LCURLY
    {switchBlockDepth++;}
    switchBlock
    {switchBlockDepth--;}
    RCURLY
  ;

switchBlock
  : switchLabeledRule+                                  #switchRules
  | groups+=switchBlockStatementGroup* emptyLabels+=switchLabel*  #switchBlocks
  ;

switchLabeledRule
  : switchLabeledExpression
  | switchLabeledBlock
  | switchLabeledThrow
  ;
```

```
switchLabeledExpression
  : switchLabel LAMBDA expression SEMI
  ;

switchLabeledBlock
  : switchLabel LAMBDA block
  ;

switchLabeledThrow
  : switchLabel LAMBDA LITERAL_THROW expression SEMI
  ;

elseStat
  : LITERAL_ELSE statement
  ;

catchClause
  : LITERAL_CATCH LPAREN catchParameter RPAREN block
  ;

catchParameter
  : mods+=variableModifier* catchType id
  ;

catchType
  : classOrInterfaceType[false] (BOR classOrInterfaceType[false])*
  ;

finallyBlock
  : LITERAL_FINALLY block
  ;

resourceSpecification
  : LPAREN resources SEMI? RPAREN
  ;

resources
  : resource (SEMI resource)*
  ;

resource
  : resourceDeclaration | variableAccess
  ;

resourceDeclaration
  : mods+=variableModifier* type=classOrInterfaceType[true]
    variableDeclaratorId[$ctx.mods, $ctx.type] ASSIGN expression
```

```
  ;

variableAccess
  : accessList+=fieldAccessNoIdent* (id | LITERAL_THIS)
  ;

fieldAccessNoIdent
  : expr DOT
  ;

/**
* Matches cases then statements, both of which are mandatory.
* To handle empty cases at the end, we add switchLabel* to statement.
*/
switchBlockStatementGroup
  : switchLabel+ slists+=blockStatement+
  ;

switchLabel
  : LITERAL_CASE caseConstants COLON?                    #caseLabel
  | LITERAL_DEFAULT COLON?                               #defaultLabel
  ;

caseConstants
  : expression (COMMA expression)*
  ;

forControl
  : LPAREN enhancedForControl RPAREN                     #enhancedFor
  | LPAREN forInit? SEMI forCond=expression?
    SEMI forUpdate=expressionList? RPAREN                #forFor
  ;

forInit
  : localVariableDeclaration
  | expressionList
  ;

enhancedForControl
  : mods+=variableModifier* type=typeType[true]
    variableDeclaratorId[$ctx.mods, $ctx.type] COLON expression
  ;

// EXPRESSIONS

parExpression
  : LPAREN expression RPAREN
  ;
```

```
expressionList
  : startExp=expression (COMMA expression)*
  ;

// We will use this rule to make sure that we have an EXPR node
// at the top of all expression subtrees.
expression
  : expr
  ;

expr
  : primary                                                    #primaryExp
  | expr bop=DOT id                                            #refOp
  | expr bop=DOT id LPAREN expressionList? RPAREN              #methodCall
  | expr bop=DOT LITERAL_THIS                                  #thisExp
  | expr bop=DOT LITERAL_NEW nonWildcardTypeArguments?
    innerCreator                                               #initExp
  | expr bop=DOT nonWildcardTypeArguments?
    LITERAL_SUPER superSuffix?                                 #superExp
  | expr bop=DOT
    nonWildcardTypeArguments
    id LPAREN expressionList? RPAREN                           #invOp
  | expr LBRACK expr RBRACK                                    #indexOp
  | id LPAREN expressionList? RPAREN                           #simpleMethodCall
  | LITERAL_NEW creator                                        #newExp
  | expr postfix=(INC | DEC)                                   #postfix
  | prefix=(PLUS | MINUS | INC | DEC) expr                     #prefix
  | prefix=(BNOT | LNOT) expr                                  #prefix
  | expr DOUBLE_COLON typeArguments? (id | LITERAL_NEW)        #methodRef
  | typeType[false] DOUBLE_COLON typeArguments? (id | LITERAL_NEW)   #methodRef
  | classType DOUBLE_COLON typeArguments? LITERAL_NEW          #methodRef
  | LPAREN typeCastParameters RPAREN expr                      #castExp
  | expr bop=(STAR | DIV | MOD) expr                           #binOp
  | expr bop=(PLUS|MINUS) expr                                 #binOp
  // handle bitwise shifts below, not in lexer
  | expr (LT LT | GT GT GT | GT GT) expr                       #bitShift
  | expr bop=LITERAL_INSTANCEOF (patternDefinition | typeType[true])   #instanceOfExp
  | expr bop=(LE | GE | GT | LT) expr                          #binOp
  | expr bop=(EQUAL | NOT_EQUAL) expr                          #binOp
  | expr bop=BAND expr                                         #binOp
  | expr bop=BXOR expr                                         #binOp
  | expr bop=BOR expr                                          #binOp
  | expr bop=LAND expr                                         #binOp
  | expr bop=LOR expr                                          #binOp
  | <assoc=right> expr bop=QUESTION expr COLON expr            #ternaryOp
  | <assoc=right> expr
    bop=(ASSIGN | PLUS_ASSIGN | MINUS_ASSIGN | STAR_ASSIGN | DIV_ASSIGN
```

```
      | BAND_ASSIGN | BOR_ASSIGN | BXOR_ASSIGN | SR_ASSIGN | BSR_ASSIGN
      | SL_ASSIGN | MOD_ASSIGN)
    expr                                        #binOp
  | lambdaExpression                            #lambdaExp
  ;

typeCastParameters
  : typeType[true] (BAND typeType[true])*
  ;

// Java8
lambdaExpression
  : lambdaParameters LAMBDA lambdaBody
  ;

// Java8
lambdaParameters
  : id                                          #singleLambdaParam
  | LPAREN formalParameterList? RPAREN          #formalLambdaParam
  | LPAREN multiLambdaParams RPAREN             #multiLambdaParam
  ;

multiLambdaParams
  : id (COMMA id)*
  ;

// Java8
lambdaBody
  : expression
  | block
  ;

primary
  : switchExpressionOrStatement                 #switchPrimary
  | LPAREN expr RPAREN                          #parenPrimary
  | LITERAL_THIS                                #tokenPrimary
  | LITERAL_SUPER                               #tokenPrimary
  | literal                                     #literalPrimary
  | id                                          #tokenPrimary
  | type=classOrInterfaceType[false] arrayDeclarator*
    DOT LITERAL_CLASS                           #classRefPrimary
  | type=primitiveType arrayDeclarator*
    DOT LITERAL_CLASS                           #primitivePrimary
  ;

classType
  : (classOrInterfaceType[false] DOT)? annotations[false] id typeArguments?
  ;
```

```
creator
  : nonWildcardTypeArguments createdName classCreatorRest
  | annotations[false] createdName
    (annotations[false] arrayCreatorRest | classCreatorRest)
  ;

createdName
  : annotations[false] id typeArgumentsOrDiamond?
     extended+=createdNameExtended*                          #createdNameObject
  | primitiveType                                  #createdNamePrimitive
  ;

createdNameExtended
  : DOT annotations[false] id typeArgumentsOrDiamond?
  ;

innerCreator
  : annotations[false] id nonWildcardTypeArgumentsOrDiamond? classCreatorRest
  ;

arrayCreatorRest
  : LBRACK
    ( RBRACK arrayDeclarator* arrayInitializer
    | expression RBRACK bracketsWithExp* arrayDeclarator*
    )
  ;

bracketsWithExp
  : annotations[false] LBRACK expression RBRACK
  ;

classCreatorRest
  : arguments classBody?
  ;

typeArgumentsOrDiamond
  : LT GT                                  #diamond
  | typeArguments                              #typeArgs
  ;

nonWildcardTypeArgumentsOrDiamond
  : LT GT                                  #nonWildcardDiamond
  | nonWildcardTypeArguments                          #nonWildcardTypeArgs
  ;

nonWildcardTypeArguments
  : LT typeArgumentsTypeList GT
```

```
  ;

// As we build a different AST for `TYPE_ARGUMENTS` than we do for simple type list
// (used in `IMPLEMENTS` and elsewhere), we have two separate rules, 'typeArgumentsTypeList'
// and 'typeList' for what is syntactically the same construct. This keeps the visitor
// implementation simpler.
typeArgumentsTypeList
  : typeType[false] (COMMA typeType[false])*
  ;

typeList
  : typeType[false] (COMMA typeType[false])*
  ;

typeType[boolean createImaginaryNode]
  : annotations[false] classOrInterfaceOrPrimitiveType arrayDeclarator*
  ;

classOrInterfaceOrPrimitiveType
  : (classOrInterfaceType[false] | primitiveType)
  ;

arrayDeclarator
  : anno=annotations[false] LBRACK RBRACK
  ;

primitiveType
  : LITERAL_BOOLEAN
  | LITERAL_CHAR
  | LITERAL_BYTE
  | LITERAL_SHORT
  | LITERAL_INT
  | LITERAL_LONG
  | LITERAL_FLOAT
  | LITERAL_DOUBLE
  | LITERAL_VOID
  ;

typeArguments
  : LT typeArgument (COMMA typeArgument)* GT
  ;

superSuffix
  : LPAREN expressionList? RPAREN                    #superSuffixSimple
  | DOT id (LPAREN expressionList? RPAREN)?          #superSuffixDot
  ;

arguments
```

```
  : LPAREN expressionList? RPAREN
  ;

patternDefinition
  : patternVariableDefinition
  ;

patternVariableDefinition
  : mods+=modifier* type=typeType[true] id
  ;

permittedSubclassesAndInterfaces
  : LITERAL_PERMITS classOrInterfaceType[false] (COMMA classOrInterfaceType[false])*
  ;

// Handle the 'keyword as identifier' problem
id  : LITERAL_RECORD
  | LITERAL_YIELD
  | LITERAL_NON_SEALED
  | LITERAL_SEALED
  | LITERAL_PERMITS
  | IDENT
  ;
```

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/grammar/java/JavaLanguageParser.g4
No license file was found, but licenses were detected in source scan.

```
// This library is free software; you can redistribute it and/or
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
// Lesser General Public License for more details.
// You should have received a copy of the GNU Lesser General Public
```

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/PackageDefHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/FullIdent.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/PatternVariableNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/XpathFilterElement.java

* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/blocks/EmptyCatchBlockCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/FileLengthCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/BeforeExecutionFileFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/BlockParentHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SuppressionCommentFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocTags.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/SingleSpaceSeparatorCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/IllegalTypeCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/FileTabCharacterCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/IntRangeFilterElement.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/MethodDefHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/utils/InlineTagUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/CheckUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/blocks/AvoidNestedBlocksCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/IndentationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/MatchSuppressor.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/UnnecessarySemicolonInTryWithResourcesCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/SimplifyBooleanExpressionCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/ParameterNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/BlockCommentPosition.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/RecordTypeParameterNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/AuditEventFormatter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/EqualsAvoidNullCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/modifier/ModifierOrderCheck.java

* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/Definitions.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/IllegalInstantiationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/AbstractSuperCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/AbstractViolationReporter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/Context.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/DoWhileHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/NewHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/CommentSuppressor.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/OuterTypeFilenameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/DefaultConfiguration.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/AbbreviationAsWordInNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/Main.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/GenericWhitespaceCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/annotation/PackageAnnotationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/RegexpMultilineCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/InnerAssignmentCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/annotation/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/PackageDeclarationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/WhitespaceAfterCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/JavadocUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/RegexpSinglelineJavaCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/AuditEventDefaultFormatter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/ScopeUtil.java

* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/CovariantEqualsCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/MemberNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/IllegalTokenCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/ConstantNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/CheckstyleException.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/CaseHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SuppressWarningsFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/InvalidJavadocTag.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/AvoidStarImportCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/meta/ModuleDetails.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/modifier/RedundantModifierCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filefilters/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/MethodNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/header/AbstractHeaderCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/MissingJavadocPackageCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SuppressionXpathSingleFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/JavaAstVisitor.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SuppressionSingleFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/EmptyLineSeparatorCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/XpathFileGeneratorAuditListener.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/DetailNodeTreeStringPrinter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/MultipleVariableDeclarationsCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/MissingJavadocTypeCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/LocalFinalVariableNameCheck.java

* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/TypeNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/AbstractJavadocCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SuppressFilterElement.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/TreeWalkerAuditEvent.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/ClassImportRule.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/ChainedPropertyUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/OneStatementPerLineCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/CatchHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/SarifLogger.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/RootModule.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/blocks/NeedBracesCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/CodeSelectorPresentation.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/ArrayTypeStyleCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/ParameterAssignmentCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/metrics/JavaNCSSCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/GlobalStatefulCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocMethodCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/UnnecessaryParenthesesCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/NoWhitespaceBeforeCaseDefaultColonCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/header/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/EqualsHashCodeCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/SwitchHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/BeforeExecutionFileFilterSet.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/ParseTreeTablePresentation.java

* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/DetailAstImpl.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/UncommentedMainCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/HiddenFieldCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/AnnotationArrayInitHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/utils/BlockTagUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/FinallyHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocPackageCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/design/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/design/DesignForExtensionCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/meta/JavadocMetadataScraper.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/IndentLevel.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/FinalParametersCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/design/InnerTypeLastCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/xpath/ElementNode.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/ThreadModeSettings.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/StringLiteralEqualityCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/TokenUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/PackageNamesLoader.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/meta/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/AvoidInlineConditionalsCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/meta/XmlMetaReader.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/FileSetCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/ConfigurationLoader.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/design/HideUtilityClassConstructorCheck.java

* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/SuperCloneCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/YieldHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/UniquePropertiesCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/MessageDispatcher.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/annotation/MissingDeprecatedCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/TypecastParenPadCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/IfHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/AbstractClassNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/ImportHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/MethodTypeParameterNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/MissingJavadocMethodCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/AuditEvent.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/AccessModifierOption.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/CommentsIndentationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/NoArrayTrailingCommaCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/ObjectBlockHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/grammar/CommentListener.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/PrimordialHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/Contextualizable.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/AvoidStaticImportCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/HtmlTag.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/ClassDefHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/modifier/package-info.java

* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/annotation/MissingOverrideCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/MatchXpathCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/MainFrame.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/CsvFilterElement.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocStyleCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/NewlineAtEndOfFileCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocBlockTagLocationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/LineSeparatorOption.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/PackageObjectFactory.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/ExecutableStatementCountCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/LineColumn.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/TagParser.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/NoFinalizerCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/OperatorWrapCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/IntMatchFilterElement.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/RecordComponentNumberCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/FilterUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/NoCodeInFileCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/UnnecessarySemicolonAfterTypeMemberDeclarationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/AbstractImportControl.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/XMLLogger.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/XpathFileGeneratorAstFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/IllegalCatchCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-

jar/com/puppycrawl/tools/checkstyle/gui/ListToTreeSelectionModelWrapper.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/IllegalIdentifierNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/EmptyStatementCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/metrics/CyclomaticComplexityCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/AnonInnerLengthCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/metrics/BooleanExpressionComplexityCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/WhitespaceAroundCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/EmptyForIteratorPadCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/HandlerFactory.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/CatchParameterNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/IllegalImportCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/FinalLocalVariableCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/StatelessCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/ImportControlLoader.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/MainFrameModel.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/VariableDeclarationUsageDistanceCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/DetectorOptions.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/LabelHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/MethodCallHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/Configuration.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/TrailingCommentCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/LambdaHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/OuterTypeNumberCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-

jar/com/puppycrawl/tools/checkstyle/PropertiesExpander.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/NeverSuppress.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/TreeWalker.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/LambdaBodyLengthCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/AvoidDoubleBraceInitializationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/design/MutableExceptionCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/DefaultLogger.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/UnnecessarySemicolonAfterOuterTypeDeclarationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/xpath/AbstractNode.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/PackageNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/ImportOrderCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/NoWhitespaceAfterCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/ant/CheckstyleAntTask.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/blocks/RightCurlyCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/xpath/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/XpathUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/modifier/ClassMemberImpliedModifierCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/grammar/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/AstTreeStringPrinter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/ParenPadCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/grammar/CrAwareLexerSimulator.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/NestedForDepthCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/MultilineDetector.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/AvoidEscapedUnicodeCharactersCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-

jar/com/puppycrawl/tools/checkstyle/api/ExternalResourceHolder.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/StaticInitHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SeverityMatchFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/CustomImportOrderCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/design/FinalClassCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/design/InterfaceIsTypeCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/MissingSwitchDefaultCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/DescendantTokenCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/design/VisibilityModifierCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/SuppressWarningsHolder.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/WhileHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/FileContents.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/TryHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocTagContinuationIndentationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/xpath/XpathQueryGenerator.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/ant/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/OverloadMethodsDeclarationOrderCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/InterfaceTypeParameterNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SuppressionsLoader.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/ForHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/Scope.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/meta/ModuleType.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/EmptyForInitializerPadCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-

jar/com/puppycrawl/tools/checkstyle/checks/naming/LambdaParameterNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/SingleLineJavadocCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/JavaParser.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/NonEmptyAtclauseDescriptionCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/annotation/SuppressWarningsCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/CheckstyleParserErrorStrategy.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/PkgImportRule.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/RegexpOnFilenameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/TokenTypes.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/NestedIfDepthCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/blocks/LeftCurlyOption.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/AbstractNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/StaticVariableNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/TodoCommentCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SuppressionFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/UnusedImportsCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/DetailNode.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/ModuleReflectionUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/FileImportControl.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/metrics/ClassFanOutComplexityCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-

jar/com/puppycrawl/tools/checkstyle/checks/coding/DefaultComesLastCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocParagraphCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/WriteTagCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/Comment.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/RequireEmptyLineBeforeBlockTagGroupCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/TextBlock.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/MethodParamPadCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/OrderedPropertiesCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/Main.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/annotation/AnnotationLocationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocVariableCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/AutomaticBean.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/blocks/LeftCurlyCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/FilterSet.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/ClassTypeParameterNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/modifier/InterfaceMemberImpliedModifierCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/DetailAstSet.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/AbstractParenPadCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/MultipleStringLiteralsCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/AccessResult.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/WrapOption.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-

jar/com/puppycrawl/tools/checkstyle/gui/TreeTableModelAdapter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/ExplicitInitializationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/SwitchRuleHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/FallThroughCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/ReturnCountCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/AbstractImportRule.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/AbstractFileSetCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SuppressWithPlainTextCommentFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/UnnecessarySemicolonInEnumerationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/xpath/RootNode.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/blocks/BlockOption.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocTagInfo.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/MemberDefHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/meta/XmlMetaWriter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/LineLengthCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocMissingWhitespaceAfterAsteriskCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/MissingCtorCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/SeverityLevelCounter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/Configurable.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/AbstractCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/JavadocPropertiesGenerator.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/Violation.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/blocks/EmptyBlockCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/TreeTable.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-

jar/com/puppycrawl/tools/checkstyle/meta/MetadataGeneratorUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/CommonUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/SuppressWithNearbyCommentFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/ModuleFactory.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/xpath/AttributeNode.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/SeparatorWrapCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/metrics/NPathComplexityCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/ArrayInitHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/ParserUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/IndexHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/design/ThrowsCountCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/TreeWalkerFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/InvalidJavadocPositionCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/JavadocDetailNodeParser.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/design/OneTopLevelClassCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/Checker.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/DefaultContext.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/ArrayTrailingCommaCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/SeverityLevel.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/SynchronizedHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/meta/ModulePropertyDetails.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/RequireThisCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/SuppressionsStringPrinter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-

jar/com/puppycrawl/tools/checkstyle/checks/javadoc/utils/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/NoLineWrapCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/MagicNumberCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/PadOption.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/ParseTreeTableModel.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/MethodLengthCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/RegexpSinglelineCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/AuditListener.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/FileStatefulCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocTag.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/regexp/SinglelineDetector.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/AvoidNoArgumentSuperConstructorCallCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/ModifiedControlVariableCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/NoCloneCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/FileText.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/SummaryJavadocCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/LineWrappingHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/AbstractExpressionHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/DeclarationOrderCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/annotation/AnnotationOnSameLineCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/Filter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/api/JavadocTokenTypes.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/utils/AnnotationUtil.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/meta/MetadataGenerationException.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-

jar/com/puppycrawl/tools/checkstyle/checks/coding/SimplifyBooleanReturnCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/annotation/AnnotationUseStyleCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/blocks/RightCurlyOption.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/blocks/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/NestedTryDepthCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/BaseCellEditor.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/IllegalTokenTextCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/CodeSelector.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/whitespace/NoWhitespaceBeforeCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocNodeImpl.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/RecordComponentNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/gui/TreeTableCellRenderer.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/LocalVariableNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/PropertyResolver.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/utils/TagInfo.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/RedundantImportCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/PkgImportControl.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filefilters/BeforeExecutionExclusionFileFilter.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/XmlLoader.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/MismatchStrategy.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/MethodCountCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/AtclauseOrderCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-

jar/com/puppycrawl/tools/checkstyle/api/DetailAST.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/IllegalThrowsCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocTypeCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/ImportControlCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/imports/ImportOrderOption.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/SlistHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/indentation/ElseHandler.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/sizes/ParameterNumberCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/TranslationCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/metrics/AbstractClassCouplingCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/metrics/ClassDataAbstractionCouplingCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/javadoc/JavadocMissingLeadingAsteriskCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/filters/IntFilterElement.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/UpperEllCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/SuperFinalizeCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/naming/AbstractAccessControlNameCheck.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/PropertyCacheFile.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/metrics/package-info.java
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/coding/NoEnumTrailingCommaCheck.java
No license file was found, but licenses were detected in source scan.

// This library is free software; you can redistribute it and/or
// modify it under the terms of the GNU Lesser General Public
// License as published by the Free Software Foundation; either
// version 2.1 of the License, or (at your option) any later version.
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
// Lesser General Public License for more details.
// You should have received a copy of the GNU Lesser General Public

```
/**
* <p>
* Checks the header of a source file against a header that contains a
* <a href="https://docs.oracle.com/en/java/javase/11/docs/api/java.base/java/util/regex/Pattern.html">
* regular expression</a> for each line of the source header.
* </p>
* <p>
* Rationale: In some projects <a href="https://checkstyle.org/config_header.html#Header">
* checking against a fixed header</a> is not sufficient, e.g. the header might
* require a copyright line where the year information is not static.
* </p>
* <p>
* For example, consider the following header:
* </p>
* <pre>
* line  1: ^/{71}$
* line  2: ^// checkstyle:$
* line  3: ^// Checks Java source code for adherence to a set of rules\.$
* line  4: ^// Copyright \(C\) \d\d\d\d  Oliver Burn$
* line  5: ^// Last modification by \$Author.*\$$
* line  6: ^/{71}$
* line  7:
* line  8: ^package
* line  9:
* line 10: ^import
* line 11:
* line 12: ^/\*\*
* line 13: ^ \*([^/]|$)
* line 14: ^ \*&#47;
* </pre>
* <p>
* Lines 1 and 6 demonstrate a more compact notation for 71 '/' characters.
* Line 4 enforces that the copyright notice includes a four digit year.
* Line 5 is an example how to enforce revision control keywords in a file header.
* Lines 12-14 is a template for javadoc (line 13 is so complicated to remove
* conflict with and of javadoc comment). Lines 7, 9 and 11 will be treated
* as '^$' and will forcefully expect the line to be empty.
* </p>
* <p>
* Different programming languages have different comment syntax rules,
* but all of them start a comment with a non-word character.
* Hence you can often use the non-word character class to abstract away
* the concrete comment syntax and allow checking the header for different
* languages with a single header definition. For example, consider the following
* header specification (note that this is not the full Apache license header):
* </p>
* <pre>
* line 1: ^#!
```

* line 2: ^<\?xml.*>$
* line 3: ^\W*$
* line 4: ^\W*Copyright 2006 The Apache Software Foundation or its licensors, as applicable\.$
* line 5: ^\W*Licensed under the Apache License, Version 2\.0 \(the "License"\);$
* line 6: ^\W*$
* </pre>
*
* Lines 1 and 2 leave room for technical header lines, e.g. the "#!/bin/sh"
* line in Unix shell scripts, or the XML file header of XML files.
* Set the multiline property to "1, 2" so these lines can be ignored for
* file types where they do no apply. Lines 3 through 6 define the actual header content.
* Note how lines 2, 4 and 5 use escapes for characters that have special regexp semantics.
*
*
* In default configuration, if header is not specified, the default value
* of header is set to null and the check does not rise any violations.
*
*
*
* Property {@code headerFile} - Specify the name of the file containing the required header.
* Type is {@code java.net.URI}.
* Default value is {@code null}.
*
*
* Property {@code charset} - Specify the character encoding to use when reading the headerFile.
* Type is {@code java.lang.String}.
* Default value is {@code the charset property of the parent
* Checker module}.
*
*
* Property {@code header} - Define the required header specified inline.
* Individual header lines must be separated by the string {@code "\n"}
* (even on platforms with a different line separator).
* For header lines containing {@code "\n\n"} checkstyle will
* forcefully expect an empty line to exist. See examples below.
* Regular expressions must not span multiple lines.
* Type is {@code java.lang.String}.
* Default value is {@code null}.
*
*
* Property {@code multiLines} - Specify the line numbers to repeat (zero or more times).
* Type is {@code int[]}.
* Default value is {@code ""}.
*
*
* Property {@code fileExtensions} - Specify the file type extension of files to process.
* Type is {@code java.lang.String[]}.
* Default value is {@code ""}.

```
* </li>
* </ul>
* <p>
* To configure the check such that no violations arise.
* Default values of properties are used.
* </p>
* <pre>
* &lt;module name="RegexpHeader"/&gt;
* </pre>
* <p>
* To configure the check to use header file {@code "config/java.header"} and
* {@code 10} and {@code 13} multi-lines:
* </p>
* <pre>
* &lt;module name="RegexpHeader"&gt;
*   &lt;property name="headerFile" value="config/java.header"/&gt;
*   &lt;property name="multiLines" value="10, 13"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* To configure the check to verify that each file starts with the header
* </p>
* <pre>
* ^// Copyright \(C\) (\d\d\d\d -)? 2004 MyCompany$
* ^// All rights reserved$
* </pre>
* <p>
* without the need for an external header file:
* </p>
* <pre>
* &lt;module name="RegexpHeader"&gt;
*   &lt;property
*     name="header"
*     value="^// Copyright \(C\) (\d\d\d\d -)? 2004 MyCompany$
*       \n^// All rights reserved$"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
* For regex containing {@code "\n\n"}
* </p>
* <pre>
* &lt;module name="RegexpHeader"&gt;
*   &lt;property
*     name="header"
*     value="^package .*\n\n.*"/&gt;
* &lt;/module&gt;
* </pre>
* <p>
```

* {@code "\n\n"} will be treated as '^$' and will forcefully expect the line
* to be empty. For example -
*
* <pre>
* package com.some.package;
* public class ThisWillFail { }
* </pre>
*
* would fail for the regex above. Expected -
*
* <pre>
* package com.some.package;
*
* public class ThisWillPass { }
* </pre>
*
* <u>Note</u>: {@code ignoreLines} property has been removed from this check to simplify it.
* To make some line optional use "^.*$" regexp for this line.
*
*
* Parent is {@code com.puppycrawl.tools.checkstyle.Checker}
*
*
* Violation Message Keys:
*
*
*
* {@code header.mismatch}
*
*
* {@code header.missing}
*
*
*
* @since 6.9
*/

Found in path(s):
* /opt/cola/permits/1340816474_1654719639.8705432/0/checkstyle-9-0-1-sources-jar/com/puppycrawl/tools/checkstyle/checks/header/RegexpHeaderCheck.java

1.194 jackson-dataformat-yaml 2.13.3

1.194.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.

It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers, as well as supported
commercially by FasterXML.com.

Licensing

Jackson core and extension components may be licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact
FasterXML.com (http://fasterxml.com).

Credits

A list of contributors may be found from CREDITS file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.
This copy of Jackson JSON processor YAML module is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.195 core-functionality-for-the-reactor-netty-library 1.0.21

1.195.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2018-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/Http2StreamBridgeServerHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2019-2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpResponseDecoderSpec.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpMetricsRecorder.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientInfos.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HAProxyMessageDetector.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/ProxyProtocolSupportType.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientMetricsRecorder.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HAProxyMessageReader.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpDecoderSpec.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/PrematureCloseException.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2011-2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/Cookies.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpPredicate.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerRequest.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/package-info.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/websocket/WebsocketInbound.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/package-info.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerRoutes.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/websocket/WebsocketOutbound.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpInfos.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientResponse.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/package-info.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientRequest.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerResponse.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/DefaultHttpServerRoutes.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2020-2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/WebsocketClientSpecImpl.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogArgProviderH1.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/BaseAccessLogHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogArgProvider.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/websocket/package-info.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/WebsocketServerSpec.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/websocket/WebsocketSpec.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLog.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/WebsocketClientSpec.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AbstractAccessLogArgProvider.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpTrafficHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogHandlerFactory.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogArgProviderH2.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/DefaultHttpForwardedHeaderHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/Http2SettingsSpec.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/package-info.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/FailedHttpClientRequest.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/websocket/WebsocketSpecImpl.java

* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogFactory.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/WebsocketServerSpecImpl.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2021-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/Http2Pool.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/AbstractHttpClientMetricsHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/MicrometerHttp2ConnectionProviderMeterRegistrar.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/ContextAwareHttpServerMetricsHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/AbstractHttpServerMetricsHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/ContextAwareHttpClientMetricsHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2018-2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientSecure.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerState.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpRequestDecoderSpec.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogHandlerH2.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/WebsocketFinalizer.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientState.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientTcpConfig.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/SimpleCompressionHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerTcpConfig.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/logging/AccessLogHandlerH1.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpProtocol.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/ConnectionInfo.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerFormDecoderProvider.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerInfos.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/NonSslRedirectDetector.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/ContextAwareHttpServerMetricsRecorder.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/ContextAwareHttpClientMetricsRecorder.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/ServerCookies.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/NonSslRedirectHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/ContextAwareHttpMetricsRecorder.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/Http2SslContextSpec.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/Http11SslContextSpec.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpRouteHandlerMetadata.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientSecurityUtils.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (c) 2017-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientConnect.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (c) 2020-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerConfig.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/Http2ConnectionProvider.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/Http2StreamBridgeClientHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientConfig.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpConnectionProvider.java
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (c) 2017-2021 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerBind.java

* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/UriEndpoint.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientFinalizer.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/UriEndpointFactory.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientForm.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2019-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientMetricsHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerMetricsHandler.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerMetricsRecorder.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/MicrometerHttpClientMetricsRecorder.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/MicrometerHttpServerMetricsRecorder.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/MicrometerHttpMetricsRecorder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*

*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/Http2AllocationStrategy.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2011-2022 VMware, Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/RedirectClientException.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClient.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpOperations.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/HttpResources.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServer.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpServerOperations.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/WebsocketServerOperations.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/HttpClientFormEncoder.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-

jar/reactor/netty/http/client/HttpClientOperations.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/client/WebsocketClientOperations.java
* /opt/cola/permits/1519137153_1671725114.6287546/0/reactor-netty-http-1-0-21-sources-jar/reactor/netty/http/server/HttpTrafficHandler.java

1.196 zstd-jni 1.5.0-4

1.196.1 Available under license :

Zstd-jni: JNI bindings to Zstd Library

Copyright (c) 2015-present, Luben Karavelov/ All rights reserved.

BSD License

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
 list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this
 list of conditions and the following disclaimer in the documentation and/or
 other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.197 antlr 4.9.2

1.197.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
[The "BSD licence"]
Copyright (c) 2006 Kay Roepke
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
/*
This file contains the actual layout of the messages emitted by ANTLR.
This file contains the default format ANTLR uses.
*/

location(file, line, column) ::= "<file>:<line>:<column>:"

message(id, text) ::= "(<id>) <text>"

report(location, message, type) ::= "<type>(<message.id>): <location> <message.text>"

wantsSingleLineMessage() ::= "false"

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/tool/templates/messages/formats/antlr.stg
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions

```
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

/** The definitive ANTLR v3 tree grammar to walk/visit ANTLR v4 grammars.
*  Parses trees created by ANTLRParser.g.
*
*  Rather than have multiple tree grammars, one for each visit, I'm
*  creating this generic visitor that knows about context. All of the
*  boilerplate pattern recognition is done here. Then, subclasses can
*  override the methods they care about. This prevents a lot of the same
*  context tracking stuff like "set current alternative for current
*  rule node" that is repeated in lots of tree filters.
*/
tree grammar GrammarTreeVisitor;
options {
 language      = Java;
 tokenVocab    = ANTLRParser;
 ASTLabelType  = GrammarAST;
}

// Include the copyright in this source and also the generated source
@header {
/*
[The "BSD license"]
Copyright (c) 2011 Terence Parr
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
```

are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
```
package org.antlr.v4.parse;
import org.antlr.v4.Tool;
import org.antlr.v4.tool.*;
import org.antlr.v4.tool.ast.*;
import java.lang.reflect.InvocationTargetException;
import java.lang.reflect.Method;
}

@members {
public String grammarName;
public GrammarAST currentRuleAST;
public String currentModeName = LexerGrammar.DEFAULT_MODE_NAME;
public String currentRuleName;
public GrammarAST currentOuterAltRoot;
public int currentOuterAltNumber = 1; // 1..n
public int rewriteEBNFLevel = 0;

public GrammarTreeVisitor() { this(null); }

// Should be abstract but can't make gen'd parser abstract;
// subclasses should implement else everything goes to stderr!
public ErrorManager getErrorManager() { return null; }

public void visitGrammar(GrammarAST t) { visit(t, "grammarSpec"); }
public void visit(GrammarAST t, String ruleName) {
 CommonTreeNodeStream nodes = new CommonTreeNodeStream(new GrammarASTAdaptor(), t);
 setTreeNodeStream(nodes);
```

```
 try {
  Method m = getClass().getMethod(ruleName);
  m.invoke(this);
 }
 catch (Throwable e) {
  ErrorManager errMgr = getErrorManager();
  if ( e instanceof InvocationTargetException ) {
   e = e.getCause();
  }
  //e.printStackTrace(System.err);
  if ( errMgr==null ) {
   System.err.println("can't find rule "+ruleName+
         " or tree structure error: "+t.toStringTree()
         );
   e.printStackTrace(System.err);
  }
  else errMgr.toolError(ErrorType.INTERNAL_ERROR, e);
 }
}

public void discoverGrammar(GrammarRootAST root, GrammarAST ID) { }
public void finishPrequels(GrammarAST firstPrequel) { }
public void finishGrammar(GrammarRootAST root, GrammarAST ID) { }

public void grammarOption(GrammarAST ID, GrammarAST valueAST) { }
public void ruleOption(GrammarAST ID, GrammarAST valueAST) { }
public void blockOption(GrammarAST ID, GrammarAST valueAST) { }
public void defineToken(GrammarAST ID) { }
public void defineChannel(GrammarAST ID) { }
public void globalNamedAction(GrammarAST scope, GrammarAST ID, ActionAST action) { }
public void importGrammar(GrammarAST label, GrammarAST ID) { }

public void modeDef(GrammarAST m, GrammarAST ID) { }

public void discoverRules(GrammarAST rules) { }
public void finishRules(GrammarAST rule) { }
public void discoverRule(RuleAST rule, GrammarAST ID, List<GrammarAST> modifiers,
      ActionAST arg, ActionAST returns, GrammarAST thrws,
      GrammarAST options, ActionAST locals,
      List<GrammarAST> actions,
      GrammarAST block) { }
public void finishRule(RuleAST rule, GrammarAST ID, GrammarAST block) { }
public void discoverLexerRule(RuleAST rule, GrammarAST ID, List<GrammarAST> modifiers,
                  GrammarAST block) { }
public void finishLexerRule(RuleAST rule, GrammarAST ID, GrammarAST block) { }
public void ruleCatch(GrammarAST arg, ActionAST action) { }
public void finallyAction(ActionAST action) { }
public void discoverOuterAlt(AltAST alt) { }
```

```
public void finishOuterAlt(AltAST alt) { }
public void discoverAlt(AltAST alt) { }
public void finishAlt(AltAST alt) { }

public void ruleRef(GrammarAST ref, ActionAST arg) { }
public void tokenRef(TerminalAST ref) { }
public void elementOption(GrammarASTWithOptions t, GrammarAST ID, GrammarAST valueAST) { }
public void stringRef(TerminalAST ref) { }
public void wildcardRef(GrammarAST ref) { }
public void actionInAlt(ActionAST action) { }
public void sempredInAlt(PredAST pred) { }
public void label(GrammarAST op, GrammarAST ID, GrammarAST element) { }
public void lexerCallCommand(int outerAltNumber, GrammarAST ID, GrammarAST arg) { }
public void lexerCommand(int outerAltNumber, GrammarAST ID) { }

protected void enterGrammarSpec(GrammarAST tree) { }
protected void exitGrammarSpec(GrammarAST tree) { }

protected void enterPrequelConstructs(GrammarAST tree) { }
protected void exitPrequelConstructs(GrammarAST tree) { }

protected void enterPrequelConstruct(GrammarAST tree) { }
protected void exitPrequelConstruct(GrammarAST tree) { }

protected void enterOptionsSpec(GrammarAST tree) { }
protected void exitOptionsSpec(GrammarAST tree) { }

protected void enterOption(GrammarAST tree) { }
protected void exitOption(GrammarAST tree) { }

protected void enterOptionValue(GrammarAST tree) { }
protected void exitOptionValue(GrammarAST tree) { }

protected void enterDelegateGrammars(GrammarAST tree) { }
protected void exitDelegateGrammars(GrammarAST tree) { }

protected void enterDelegateGrammar(GrammarAST tree) { }
protected void exitDelegateGrammar(GrammarAST tree) { }

protected void enterTokensSpec(GrammarAST tree) { }
protected void exitTokensSpec(GrammarAST tree) { }

protected void enterTokenSpec(GrammarAST tree) { }
protected void exitTokenSpec(GrammarAST tree) { }

protected void enterChannelsSpec(GrammarAST tree) { }
protected void exitChannelsSpec(GrammarAST tree) { }
```

```
protected void enterChannelSpec(GrammarAST tree) { }
protected void exitChannelSpec(GrammarAST tree) { }

protected void enterAction(GrammarAST tree) { }
protected void exitAction(GrammarAST tree) { }

protected void enterRules(GrammarAST tree) { }
protected void exitRules(GrammarAST tree) { }

protected void enterMode(GrammarAST tree) { }
protected void exitMode(GrammarAST tree) { }

protected void enterLexerRule(GrammarAST tree) { }
protected void exitLexerRule(GrammarAST tree) { }

protected void enterRule(GrammarAST tree) { }
protected void exitRule(GrammarAST tree) { }

protected void enterExceptionGroup(GrammarAST tree) { }
protected void exitExceptionGroup(GrammarAST tree) { }

protected void enterExceptionHandler(GrammarAST tree) { }
protected void exitExceptionHandler(GrammarAST tree) { }

protected void enterFinallyClause(GrammarAST tree) { }
protected void exitFinallyClause(GrammarAST tree) { }

protected void enterLocals(GrammarAST tree) { }
protected void exitLocals(GrammarAST tree) { }

protected void enterRuleReturns(GrammarAST tree) { }
protected void exitRuleReturns(GrammarAST tree) { }

protected void enterThrowsSpec(GrammarAST tree) { }
protected void exitThrowsSpec(GrammarAST tree) { }

protected void enterRuleAction(GrammarAST tree) { }
protected void exitRuleAction(GrammarAST tree) { }

protected void enterRuleModifier(GrammarAST tree) { }
protected void exitRuleModifier(GrammarAST tree) { }

protected void enterLexerRuleBlock(GrammarAST tree) { }
protected void exitLexerRuleBlock(GrammarAST tree) { }

protected void enterRuleBlock(GrammarAST tree) { }
protected void exitRuleBlock(GrammarAST tree) { }
```

```
protected void enterLexerOuterAlternative(AltAST tree) { }
protected void exitLexerOuterAlternative(AltAST tree) { }

protected void enterOuterAlternative(AltAST tree) { }
protected void exitOuterAlternative(AltAST tree) { }

protected void enterLexerAlternative(GrammarAST tree) { }
protected void exitLexerAlternative(GrammarAST tree) { }

protected void enterLexerElements(GrammarAST tree) { }
protected void exitLexerElements(GrammarAST tree) { }

protected void enterLexerElement(GrammarAST tree) { }
protected void exitLexerElement(GrammarAST tree) { }

protected void enterLabeledLexerElement(GrammarAST tree) { }
protected void exitLabeledLexerElement(GrammarAST tree) { }

protected void enterLexerBlock(GrammarAST tree) { }
protected void exitLexerBlock(GrammarAST tree) { }

protected void enterLexerAtom(GrammarAST tree) { }
protected void exitLexerAtom(GrammarAST tree) { }

protected void enterActionElement(GrammarAST tree) { }
protected void exitActionElement(GrammarAST tree) { }

protected void enterAlternative(AltAST tree) { }
protected void exitAlternative(AltAST tree) { }

protected void enterLexerCommand(GrammarAST tree) { }
protected void exitLexerCommand(GrammarAST tree) { }

protected void enterLexerCommandExpr(GrammarAST tree) { }
protected void exitLexerCommandExpr(GrammarAST tree) { }

protected void enterElement(GrammarAST tree) { }
protected void exitElement(GrammarAST tree) { }

protected void enterAstOperand(GrammarAST tree) { }
protected void exitAstOperand(GrammarAST tree) { }

protected void enterLabeledElement(GrammarAST tree) { }
protected void exitLabeledElement(GrammarAST tree) { }

protected void enterSubrule(GrammarAST tree) { }
protected void exitSubrule(GrammarAST tree) { }
```

```
protected void enterLexerSubrule(GrammarAST tree) { }
protected void exitLexerSubrule(GrammarAST tree) { }

protected void enterBlockSuffix(GrammarAST tree) { }
protected void exitBlockSuffix(GrammarAST tree) { }

protected void enterEbnfSuffix(GrammarAST tree) { }
protected void exitEbnfSuffix(GrammarAST tree) { }

protected void enterAtom(GrammarAST tree) { }
protected void exitAtom(GrammarAST tree) { }

protected void enterBlockSet(GrammarAST tree) { }
protected void exitBlockSet(GrammarAST tree) { }

protected void enterSetElement(GrammarAST tree) { }
protected void exitSetElement(GrammarAST tree) { }

protected void enterBlock(GrammarAST tree) { }
protected void exitBlock(GrammarAST tree) { }

protected void enterRuleref(GrammarAST tree) { }
protected void exitRuleref(GrammarAST tree) { }

protected void enterRange(GrammarAST tree) { }
protected void exitRange(GrammarAST tree) { }

protected void enterTerminal(GrammarAST tree) { }
protected void exitTerminal(GrammarAST tree) { }

protected void enterElementOptions(GrammarAST tree) { }
protected void exitElementOptions(GrammarAST tree) { }

protected void enterElementOption(GrammarAST tree) { }
protected void exitElementOption(GrammarAST tree) { }

 @Override
 public void traceIn(String ruleName, int ruleIndex)  {
  System.err.println("enter "+ruleName+": "+input.LT(1));
 }

 @Override
 public void traceOut(String ruleName, int ruleIndex)  {
  System.err.println("exit "+ruleName+": "+input.LT(1));
 }
}
```

grammarSpec

```
@init {
 enterGrammarSpec($start);
}
@after {
 exitGrammarSpec($start);
}
  :   ^( GRAMMAR ID {grammarName=$ID.text;}
     {discoverGrammar((GrammarRootAST)$GRAMMAR, $ID);}
      prequelConstructs
     {finishPrequels($prequelConstructs.firstOne);}
      rules mode*
     {finishGrammar((GrammarRootAST)$GRAMMAR, $ID);}
  )
 ;

prequelConstructs returns [GrammarAST firstOne=null]
@init {
 enterPrequelConstructs($start);
}
@after {
 exitPrequelConstructs($start);
}
 : {$firstOne=$start;} prequelConstruct+
 |
 ;

prequelConstruct
@init {
 enterPrequelConstructs($start);
}
@after {
 exitPrequelConstructs($start);
}
 :   optionsSpec
   |   delegateGrammars
   |   tokensSpec
   |   channelsSpec
   |   action
   ;

optionsSpec
@init {
 enterOptionsSpec($start);
}
@after {
 exitOptionsSpec($start);
}
 : ^(OPTIONS option*)
```

```
  ;

option
@init {
 enterOption($start);
 boolean rule = inContext("RULE ...");
 boolean block = inContext("BLOCK ...");
}
@after {
 exitOption($start);
}
  :  ^(a=ASSIGN ID v=optionValue)
   {
   if ( block ) blockOption($ID, $v.start); // most specific first
   else if ( rule ) ruleOption($ID, $v.start);
   else grammarOption($ID, $v.start);
   }
  ;

optionValue returns [String v]
@init {
 enterOptionValue($start);
 $v = $start.token.getText();
}
@after {
 exitOptionValue($start);
}
  :  ID
  |  STRING_LITERAL
  |  INT
  ;

delegateGrammars
@init {
 enterDelegateGrammars($start);
}
@after {
 exitDelegateGrammars($start);
}
 :  ^(IMPORT delegateGrammar+)
 ;

delegateGrammar
@init {
 enterDelegateGrammar($start);
}
@after {
 exitDelegateGrammar($start);
```

```
}
  :  ^(ASSIGN label=ID id=ID) {importGrammar($label, $id);}
  |  id=ID      {importGrammar(null, $id);}
  ;

tokensSpec
@init {
 enterTokensSpec($start);
}
@after {
 exitTokensSpec($start);
}
 :  ^(TOKENS_SPEC tokenSpec+)
 ;

tokenSpec
@init {
 enterTokenSpec($start);
}
@after {
 exitTokenSpec($start);
}
 : ID     {defineToken($ID);}
 ;

channelsSpec
@init {
 enterChannelsSpec($start);
}
@after {
 exitChannelsSpec($start);
}
 :  ^(CHANNELS channelSpec+)
 ;

channelSpec
@init {
 enterChannelSpec($start);
}
@after {
 exitChannelSpec($start);
}
 : ID     {defineChannel($ID);}
 ;

action
@init {
 enterAction($start);
```

```
}
@after {
 exitAction($start);
}
 : ^(AT sc=ID? name=ID ACTION) {globalNamedAction($sc, $name, (ActionAST)$ACTION);}
 ;

rules
@init {
 enterRules($start);
}
@after {
 exitRules($start);
}
   : ^(RULES {discoverRules($RULES);} (rule|lexerRule)* {finishRules($RULES);})
   ;

mode
@init {
 enterMode($start);
}
@after {
 exitMode($start);
}
 : ^( MODE ID {currentModeName=$ID.text; modeDef($MODE, $ID);} lexerRule* )
 ;

lexerRule
@init {
 enterLexerRule($start);
 List<GrammarAST> mods = new ArrayList<GrammarAST>();
 currentOuterAltNumber=0;
}
@after {
 exitLexerRule($start);
}
 : ^( RULE TOKEN_REF
   {currentRuleName=$TOKEN_REF.text; currentRuleAST=$RULE;}
   (^(RULEMODIFIERS m=FRAGMENT {mods.add($m);}))?
      {discoverLexerRule((RuleAST)$RULE, $TOKEN_REF, mods, (GrammarAST)input.LT(1));}
      lexerRuleBlock
      {
      finishLexerRule((RuleAST)$RULE, $TOKEN_REF, $lexerRuleBlock.start);
      currentRuleName=null; currentRuleAST=null;
      }
      )
 ;
```

```
rule
@init {
 enterRule($start);
 List<GrammarAST> mods = new ArrayList<GrammarAST>();
 List<GrammarAST> actions = new ArrayList<GrammarAST>(); // track roots
 currentOuterAltNumber=0;
}
@after {
 exitRule($start);
}
 :   ^( RULE RULE_REF {currentRuleName=$RULE_REF.text; currentRuleAST=$RULE;}
   (^(RULEMODIFIERS (m=ruleModifier{mods.add($m.start);})+))?
   ARG_ACTION?
      ret=ruleReturns?
      thr=throwsSpec?
      loc=locals?
      ( opts=optionsSpec
     |   a=ruleAction {actions.add($a.start);}
     )*
      {discoverRule((RuleAST)$RULE, $RULE_REF, mods, (ActionAST)$ARG_ACTION,
          $ret.start!=null?(ActionAST)$ret.start.getChild(0):null,
          $thr.start, $opts.start,
          $loc.start!=null?(ActionAST)$loc.start.getChild(0):null,
          actions, (GrammarAST)input.LT(1));}
      ruleBlock exceptionGroup
      {finishRule((RuleAST)$RULE, $RULE_REF, $ruleBlock.start); currentRuleName=null; currentRuleAST=null;}
      )
   ;

exceptionGroup
@init {
 enterExceptionGroup($start);
}
@after {
 exitExceptionGroup($start);
}
   : exceptionHandler* finallyClause?
   ;

exceptionHandler
@init {
 enterExceptionHandler($start);
}
@after {
 exitExceptionHandler($start);
}
 : ^(CATCH ARG_ACTION ACTION) {ruleCatch($ARG_ACTION, (ActionAST)$ACTION);}
 ;
```

```
finallyClause
@init {
 enterFinallyClause($start);
}
@after {
 exitFinallyClause($start);
}
 : ^(FINALLY ACTION)    {finallyAction((ActionAST)$ACTION);}
 ;

locals
@init {
 enterLocals($start);
}
@after {
 exitLocals($start);
}
 : ^(LOCALS ARG_ACTION)
 ;

ruleReturns
@init {
 enterRuleReturns($start);
}
@after {
 exitRuleReturns($start);
}
 : ^(RETURNS ARG_ACTION)
 ;

throwsSpec
@init {
 enterThrowsSpec($start);
}
@after {
 exitThrowsSpec($start);
}
   : ^(THROWS ID+)
   ;

ruleAction
@init {
 enterRuleAction($start);
}
@after {
 exitRuleAction($start);
}
```

```
 : ^(AT ID ACTION)
 ;

ruleModifier
@init {
 enterRuleModifier($start);
}
@after {
 exitRuleModifier($start);
}
  : PUBLIC
  | PRIVATE
  | PROTECTED
  | FRAGMENT
  ;

lexerRuleBlock
@init {
 enterLexerRuleBlock($start);
}
@after {
 exitLexerRuleBlock($start);
}
  : ^( BLOCK
    ( {
     currentOuterAltRoot = (GrammarAST)input.LT(1);
    currentOuterAltNumber++;
    }
     lexerOuterAlternative
    )+
   )
  ;

ruleBlock
@init {
 enterRuleBlock($start);
}
@after {
 exitRuleBlock($start);
}
  : ^( BLOCK
    ( {
     currentOuterAltRoot = (GrammarAST)input.LT(1);
    currentOuterAltNumber++;
    }
     outerAlternative
    )+
   )
```

```
  ;

lexerOuterAlternative
@init {
 enterLexerOuterAlternative((AltAST)$start);
 discoverOuterAlt((AltAST)$start);
}
@after {
 finishOuterAlt((AltAST)$start);
 exitLexerOuterAlternative((AltAST)$start);
}
 : lexerAlternative
 ;


outerAlternative
@init {
 enterOuterAlternative((AltAST)$start);
 discoverOuterAlt((AltAST)$start);
}
@after {
 finishOuterAlt((AltAST)$start);
 exitOuterAlternative((AltAST)$start);
}
 : alternative
 ;

lexerAlternative
@init {
 enterLexerAlternative($start);
}
@after {
 exitLexerAlternative($start);
}
 : ^(LEXER_ALT_ACTION lexerElements lexerCommand+)
  | lexerElements
  ;

lexerElements
@init {
 enterLexerElements($start);
}
@after {
 exitLexerElements($start);
}
  : ^(ALT lexerElement+)
  ;
```

```
lexerElement
@init {
 enterLexerElement($start);
}
@after {
 exitLexerElement($start);
}
 : labeledLexerElement
 | lexerAtom
 | lexerSubrule
 |   ACTION      {actionInAlt((ActionAST)$ACTION);}
 |   SEMPRED      {sempredInAlt((PredAST)$SEMPRED);}
 |   ^(ACTION elementOptions) {actionInAlt((ActionAST)$ACTION);}
 |   ^(SEMPRED elementOptions) {sempredInAlt((PredAST)$SEMPRED);}
 | EPSILON
 ;

labeledLexerElement
@init {
 enterLabeledLexerElement($start);
}
@after {
 exitLabeledLexerElement($start);
}
   :   ^((ASSIGN|PLUS_ASSIGN) ID (lexerAtom|block))
 ;

lexerBlock
@init {
 enterLexerBlock($start);
}
@after {
 exitLexerBlock($start);
}
 : ^(BLOCK optionsSpec? lexerAlternative+)
   ;

lexerAtom
@init {
 enterLexerAtom($start);
}
@after {
 exitLexerAtom($start);
}
   :   terminal
   |   ^(NOT blockSet)
   |   blockSet
   |   ^(WILDCARD elementOptions)
```

```
  |   WILDCARD
  | LEXER_CHAR_SET
  |   range
  |   ruleref
  ;

actionElement
@init {
 enterActionElement($start);
}
@after {
 exitActionElement($start);
}
 : ACTION
 |   ^(ACTION elementOptions)
 |   SEMPRED
 |   ^(SEMPRED elementOptions)
 ;

alternative
@init {
 enterAlternative((AltAST)$start);
 discoverAlt((AltAST)$start);
}
@after {
 finishAlt((AltAST)$start);
 exitAlternative((AltAST)$start);
}
 : ^(ALT elementOptions? element+)
 | ^(ALT elementOptions? EPSILON)
  ;

lexerCommand
@init {
 enterLexerCommand($start);
}
@after {
 exitLexerCommand($start);
}
 : ^(LEXER_ACTION_CALL ID lexerCommandExpr)
      {lexerCallCommand(currentOuterAltNumber, $ID, $lexerCommandExpr.start);}
 | ID
      {lexerCommand(currentOuterAltNumber, $ID);}
 ;

lexerCommandExpr
@init {
 enterLexerCommandExpr($start);
```

```
}
@after {
 exitLexerCommandExpr($start);
}
 : ID
 | INT
 ;

element
@init {
 enterElement($start);
}
@after {
 exitElement($start);
}
 : labeledElement
 | atom
 | subrule
 |   ACTION      {actionInAlt((ActionAST)$ACTION);}
 |   SEMPRED     {sempredInAlt((PredAST)$SEMPRED);}
 |   ^(ACTION elementOptions) {actionInAlt((ActionAST)$ACTION);}
 |   ^(SEMPRED elementOptions) {sempredInAlt((PredAST)$SEMPRED);}
 | range
 | ^(NOT blockSet)
 | ^(NOT block)
 ;

astOperand
@init {
 enterAstOperand($start);
}
@after {
 exitAstOperand($start);
}
 : atom
 | ^(NOT blockSet)
 | ^(NOT block)
 ;

labeledElement
@init {
 enterLabeledElement($start);
}
@after {
 exitLabeledElement($start);
}
 : ^((ASSIGN|PLUS_ASSIGN) ID element) {label($start, $ID, $element.start);}
 ;
```

```
subrule
@init {
 enterSubrule($start);
}
@after {
 exitSubrule($start);
}
 : ^(blockSuffix block)
 | block
   ;

lexerSubrule
@init {
 enterLexerSubrule($start);
}
@after {
 exitLexerSubrule($start);
}
 : ^(blockSuffix lexerBlock)
 | lexerBlock
   ;

blockSuffix
@init {
 enterBlockSuffix($start);
}
@after {
 exitBlockSuffix($start);
}
   : ebnfSuffix
   ;

ebnfSuffix
@init {
 enterEbnfSuffix($start);
}
@after {
 exitEbnfSuffix($start);
}
 : OPTIONAL
  | CLOSURE
   | POSITIVE_CLOSURE
 ;

atom
@init {
 enterAtom($start);
```

```
}
@after {
 exitAtom($start);
}
 : ^(DOT ID terminal)
 | ^(DOT ID ruleref)
   | ^(WILDCARD elementOptions) {wildcardRef($WILDCARD);}
   | WILDCARD     {wildcardRef($WILDCARD);}
   |   terminal
   | blockSet
   |   ruleref
   ;

blockSet
@init {
 enterBlockSet($start);
}
@after {
 exitBlockSet($start);
}
 : ^(SET setElement+)
 ;

setElement
@init {
 enterSetElement($start);
}
@after {
 exitSetElement($start);
}
 : ^(STRING_LITERAL elementOptions)    {stringRef((TerminalAST)$STRING_LITERAL);}
 | ^(TOKEN_REF elementOptions)   {tokenRef((TerminalAST)$TOKEN_REF);}
 | STRING_LITERAL                {stringRef((TerminalAST)$STRING_LITERAL);}
 | TOKEN_REF                     {tokenRef((TerminalAST)$TOKEN_REF);}
 | ^(RANGE a=STRING_LITERAL b=STRING_LITERAL)
  {
  stringRef((TerminalAST)$a);
  stringRef((TerminalAST)$b);
  }
     |      LEXER_CHAR_SET
 ;

block
@init {
 enterBlock($start);
}
@after {
 exitBlock($start);
```

```
}
  : ^(BLOCK optionsSpec? ruleAction* ACTION? alternative+)
  ;

ruleref
@init {
 enterRuleref($start);
}
@after {
 exitRuleref($start);
}
  : ^(RULE_REF arg=ARG_ACTION? elementOptions?)
   {
   ruleRef($RULE_REF, (ActionAST)$ARG_ACTION);
   if ( $arg!=null ) actionInAlt((ActionAST)$arg);
   }
  ;

range
@init {
 enterRange($start);
}
@after {
 exitRange($start);
}
  : ^(RANGE STRING_LITERAL STRING_LITERAL)
  ;

terminal
@init {
 enterTerminal($start);
}
@after {
 exitTerminal($start);
}
  :  ^(STRING_LITERAL elementOptions)
        {stringRef((TerminalAST)$STRING_LITERAL);}
  | STRING_LITERAL    {stringRef((TerminalAST)$STRING_LITERAL);}
  | ^(TOKEN_REF elementOptions) {tokenRef((TerminalAST)$TOKEN_REF);}
  | TOKEN_REF       {tokenRef((TerminalAST)$TOKEN_REF);}
  ;

elementOptions
@init {
 enterElementOptions($start);
}
@after {
 exitElementOptions($start);
```

```
}
  : ^(ELEMENT_OPTIONS elementOption[(GrammarASTWithOptions)$start.getParent()]*)
  ;

elementOption[GrammarASTWithOptions t]
@init {
 enterElementOption($start);
}
@after {
 exitElementOption($start);
}
  : ID       {elementOption(t, $ID, null);}
  | ^(ASSIGN id=ID v=ID)   {elementOption(t, $id, $v);}
  | ^(ASSIGN ID v=STRING_LITERAL) {elementOption(t, $ID, $v);}
  | ^(ASSIGN ID v=ACTION)   {elementOption(t, $ID, $v);}
  | ^(ASSIGN ID v=INT)    {elementOption(t, $ID, $v);}
  ;
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/GrammarTreeVisitor.g
No license file was found, but licenses were detected in source scan.


```
/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  Copyright (c) 2014 Eric Vergnaud
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
```

```
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

/** ANTLR tool checks output templates are compatible with tool code generation.
*  For now, a simple string match used on x.y of x.y.z scheme.
*  Must match Tool.VERSION during load to templates.
*
*  REQUIRED.
*/

pythonTypeInitMap ::= [
 "bool":"False",
 "int":"0",
 "float":"0.0",
 "str":"",
 default:"None" // anything other than a primitive type is an object
]

// args must be <object-model-object>, <fields-resulting-in-STs>

ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
# encoding: utf-8
from antlr4 import *
from io import StringIO
import sys
if sys.version_info[1] > 5:
 from typing import TextIO
else:
 from typing.io import TextIO

<namedActions.header>
<parser>

>>

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
from antlr4 import *
if __name__ is not None and "." in __name__:
   from .<file.parserName> import <file.parserName>
else:
   from <file.parserName> import <file.parserName>
<header>
```

```
# This class defines a complete listener for a parse tree produced by <file.parserName>.
class <file.grammarName>Listener(ParseTreeListener):

  <file.listenerNames:{lname |
# Enter a parse tree produced by <file.parserName>#<lname>.
def enter<lname; format="cap">(self, ctx:<file.parserName>.<lname; format="cap">Context):
  pass

# Exit a parse tree produced by <file.parserName>#<lname>.
def exit<lname; format="cap">(self, ctx:<file.parserName>.<lname; format="cap">Context):
  pass

}; separator="\n">

del <file.parserName>
>>


VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
from antlr4 import *
if __name__ is not None and "." in __name__:
  from .<file.parserName> import <file.parserName>
else:
  from <file.parserName> import <file.parserName>
<header>

# This class defines a complete generic visitor for a parse tree produced by <file.parserName>.

class <file.grammarName>Visitor(ParseTreeVisitor):

  <file.visitorNames:{lname |
# Visit a parse tree produced by <file.parserName>#<lname>.
def visit<lname; format="cap">(self, ctx:<file.parserName>.<lname; format="cap">Context):
  return self.visitChildren(ctx)

}; separator="\n">

del <file.parserName>
>>


fileHeader(grammarFileName, ANTLRVersion) ::= <<
# Generated from <grammarFileName> by ANTLR <ANTLRVersion>
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
```

```
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
<if(superClass)>
if __name__ is not None and "." in __name__:
  from .<superClass> import <superClass>
else:
  from <superClass> import <superClass>

<endif>
<atn>

class <parser.name> ( <if(superClass)><superClass><else>Parser<endif> ):

  grammarFileName = "<parser.grammarFileName>"

  atn = ATNDeserializer().deserialize(serializedATN())

  decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]

  sharedContextCache = PredictionContextCache()

  literalNames = [ <parser.literalNames:{t | <t>}; null="\"\<INVALID>\"", separator=", ", wrap, anchor> ]

  symbolicNames = [ <parser.symbolicNames:{t | <t>}; null="\"\<INVALID>\"", separator=", ", wrap, anchor> ]

  <if(parser.rules)>
  <parser.rules:{r | RULE_<r.name> = <r.index>}; separator="\n", wrap, anchor>
  <endif>

  ruleNames =  [ <parser.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor> ]

  EOF = <TokenLabelType()>.EOF
  <if(parser.tokens)>
  <parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator="\n", wrap, anchor>
  <endif>

  <parser:(ctor)()>

  <namedActions.members>

  <funcs; separator="\n">


<if(sempredFuncs)>
  def sempred(self, localctx:RuleContext, ruleIndex:int, predIndex:int):
      if self._predicates == None:
          self._predicates = dict()
```

```
<parser.sempredFuncs.values:{f |
      self._predicates[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n        ">
      pred = self._predicates.get(ruleIndex, None)
      if pred is None:
         raise Exception("No predicate with index:" + str(ruleIndex))
      else:
         return pred(localctx, predIndex)

  <sempredFuncs.values; separator="\n">
<endif>



>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
def action(self, localctx:RuleContext, ruleIndex:int, actionIndex:int):
  if self._actions is None:
      actions = dict()
<recog.actionFuncs.values:{f|
      actions[<f.ruleIndex>] = self.<f.name>_action }; separator="\n">
      self._actions = actions
  action = self._actions.get(ruleIndex, None)
  if action is not None:
      action(localctx, actionIndex)
  else:
      raise Exception("No registered action for:" + str(ruleIndex))

<actionFuncs.values; separator="\n">

<endif>
<if(sempredFuncs)>
def sempred(self, localctx:RuleContext, ruleIndex:int, predIndex:int):
  if self._predicates is None:
      preds = dict()
<recog.sempredFuncs.values:{f|
      preds[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n">
      self._predicates = preds
  pred = self._predicates.get(ruleIndex, None)
  if pred is not None:
      return pred(localctx, predIndex)
  else:
      raise Exception("No registered predicate for:" + str(ruleIndex))

<sempredFuncs.values; separator="\n">
<endif>
>>
```

```
parser_ctor(p) ::= <<
def __init__(self, input:TokenStream, output:TextIO = sys.stdout):
  super().__init__(input, output)
  self.checkVersion("<file.ANTLRVersion>")
  self._interp = ParserATNSimulator(self, self.atn, self.decisionsToDFA, self.sharedContextCache)
  self._predicates = None

>>

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<

def <r.name>_action(self, localctx:<r.ctxType> , actionIndex:int):
<actions:{index|
<if(first(actions))>
  if actionIndex == <index>:
      <actions.(index)>
<elseif(rest(actions))>
  elif actionIndex == <index>:
      <actions.(index)>
<endif> }; separator="\n">
>>

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
def <r.name>_sempred(self, localctx:<r.ctxType>, predIndex:int):
  <actions:{index|
<if(first(actions))>
  if predIndex == <index>:
      return <actions.(index)>
<elseif(rest(actions))>
  elif predIndex == <index>:
      return <actions.(index)>
<endif> }; separator="\n">

>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<ruleCtx>

<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">
```

```
def <currentRule.name>(self<currentRule.args:{a | , <a.name><if(a.type)>:<a.type><endif>}>):

  localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, self.state<currentRule.args:{a | , <a.name>}>)
  self.enterRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>)
  <namedActions.init>
  <locals; separator="\n">
  try:
     <code>
     <postamble; separator="\n">
     <namedActions.after>
  <if(exceptions)>
  <exceptions; separator="\n">
  <else>
  except RecognitionException as re:
     localctx.exception = re
     self._errHandler.reportError(self, re)
     self._errHandler.recover(self, re)
  <endif>
  finally:
     <finallyAction>
     self.exitRule()
  return localctx

>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
 namedActions,finallyAction,postamble) ::=
<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

def <currentRule.name>(self, _p:int=0<if(currentRule.args)>, <args:{a | , <a>}><endif>):
  _parentctx = self._ctx
  _parentState = self.state
  localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, _parentState<args:{a | , <a.name>}>)
  _prevctx = localctx
  _startState = <currentRule.startState>
  self.enterRecursionRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>, _p)
  <namedActions.init>
  <locals; separator="\n">
  try:
     <code>
     <postamble; separator="\n">
     <namedActions.after>
  except RecognitionException as re:
     localctx.exception = re
```

```
      self._errHandler.reportError(self, re)
      self._errHandler.recover(self, re)
  finally:
      <finallyAction>
      self.unrollRecursionContexts(_parentctx)
  return localctx

>>

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>localctx = <parser.name>.<currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(self, localctx)<endif>
self.enterOuterAlt(localctx, <currentOuterMostAltCodeBlock.alt.altNum>)
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>

LL1AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  pass}; separator="\nel">
else:
  <error>

>>

LL1OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  pass}; separator="\nel">
else:
  pass
>>

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
```

```
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
if <expr>:
  <alts; separator="\n">

<!else if ( !(<followExpr>) ) <error>!>
>>


LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
while <loopExpr>:
  <alts; separator="\n">
  self.state = <choice.loopBackStateNumber>
  self._errHandler.sync(self)
  <iteration>

>>

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
<preamble; separator="\n">
while True:
  <alts; separator="\n">
  self.state = <choice.stateNumber> <! loopback/exit decision !>
  self._errHandler.sync(self)
  <iteration>
  if not (<loopExpr>):
     break

>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{alt |
if la_ == <i>:
  <alt>
  pass
```

```
}; separator="\nel">

>>

OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{alt |
if la_ == <i><if(!choice.ast.greedy)>+1<endif>:
  <alt>
}; separator="\nel">

>>

StarBlock(choice, alts, sync, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
_alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
  if _alt==1<if(!choice.ast.greedy)>+1<endif>:
    <iteration>
    <alts> <! should only be one !>
  self.state = <choice.loopBackStateNumber>
  self._errHandler.sync(self)
  _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)

>>

PlusBlock(choice, alts, error) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
_alt = 1<if(!choice.ast.greedy)>+1<endif>
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
  <alts:{alt|
if _alt == <i><if(!choice.ast.greedy)>+1<endif>:
  <alt>
}; separator="\nel">
  else:
    <error>
  self.state = <choice.loopBackStateNumber> <! loopback/exit decision !>
  self._errHandler.sync(self)
  _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)

>>

Sync(s) ::= "sync(<s.expecting.name>)"
```

```
ThrowNoViableAlt(t) ::= "raise NoViableAltException(self)"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" or ">
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShiftVar(s.varName, bits.shift)>})> and ((1 \<\< <offsetShiftVar(s.varName,
bits.shift)>) & (<bits.ttypes:{ttype | (1 \<\< <offsetShiftType(ttype, bits.shift)>)}; separator=" | ">)) != 0)
%>

isZero ::= [
"0":true,
default:false
]

offsetShiftVar(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(<shiftAmount> - <offset>)<else><shiftAmount><endif>
%>

offsetShiftType(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(<parser.name>.<shiftAmount> - <offset>)<else><parser.name>.<shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | <s.varName>==<parser.name>.<ttype>}; separator=" or ">
%>

cases(ttypes) ::= <<
if token in [<ttypes:{t | <parser.name>.<t>}; separator=", ">]:
>>

InvokeRule(r, argExprsChunks) ::= <<
self.state = <r.stateNumber>
<if(r.labels)><r.labels:{l | <labelref(l)> =
}><endif>self.<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChu
nks>)
>>

MatchToken(m) ::= <<
```

```
self.state = <m.stateNumber>
<if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>self.match(<parser.name>.<m.name>)
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels:{l | <labelref(l)> = }>self._input.LT(1)<endif>
<capture>
<if(invert)>if <m.varName> \<= 0 or <expr><else>if not(<expr>)<endif>:
<if(m.labels)><m.labels:{l |     <labelref(l)> = }><else>    <endif>self._errHandler.recoverInline(self)
else:
  self._errHandler.reportMatch(self)
  self.consume()
>>

Wildcard(w) ::= <<
self.state = <w.stateNumber>
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>self.matchWildcard()
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
self.state = <p.stateNumber>
if not <chunks>:
  from antlr4.error.Errors import FailedPredicateException
  raise FailedPredicateException(self, <p.predicate><if(failChunks)>, <failChunks><elseif(p.msg)>,
<p.msg><endif>)
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
except <catchArg>:
  <catchAction>
>>

// lexer actions are not associated with model objects

LexerSkipCommand()  ::= "skip()"
LexerMoreCommand()  ::= "more()"
LexerPopModeCommand() ::= "popMode()"
```

```
LexerTypeCommand(arg, grammar)      ::= "_type = <arg>"
LexerChannelCommand(arg, grammar)   ::= "_channel = <arg>"
LexerModeCommand(arg, grammar)      ::= "_mode = <arg>"
LexerPushModeCommand(arg, grammar)  ::= "pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "localctx.<a.name>"
LocalRef(a) ::= "localctx.<a.name>"
RetValueRef(a) ::= "localctx.<a.name>"
QRetValueRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(None if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.text)"
TokenPropertyRef_type(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.type)"
TokenPropertyRef_line(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.line)"
TokenPropertyRef_pos(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.column)"
TokenPropertyRef_channel(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.channel)"
TokenPropertyRef_index(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.tokenIndex)"
TokenPropertyRef_int(t) ::= "(0 if <ctx(t)>.<t.label> is None else int(<ctx(t)>.<t.label>.text))"

RulePropertyRef_start(r) ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.start)"
RulePropertyRef_stop(r)  ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.stop)"
RulePropertyRef_text(r)  ::= "(None if <ctx(r)>.<r.label> is None else
self._input.getText(<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop))"
RulePropertyRef_ctx(r)  ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r)  ::= "self"

ThisRulePropertyRef_start(r) ::= "localctx.start"
ThisRulePropertyRef_stop(r)  ::= "localctx.stop"
ThisRulePropertyRef_text(r)  ::= "self._input.getText(localctx.start, self._input.LT(-1))"
ThisRulePropertyRef_ctx(r)  ::= "localctx"
ThisRulePropertyRef_parser(r)  ::= "self"

NonLocalAttrRef(s)     ::= "self.getInvokingContext(<s.ruleIndex>).<s.name>"
SetNonLocalAttr(s, rhsChunks)  ::= "self.getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.append(<labelref(a.label)>)"

TokenDecl(t) ::= "self.<t.name> = None # <TokenLabelType()>"
```

```
TokenTypeDecl(t) ::= "self.<t.name> = 0 # <TokenLabelType()> type"
TokenListDecl(t) ::= "self.<t.name> = list() # of <TokenLabelType()>s"
RuleContextDecl(r) ::= "self.<r.name> = None # <r.ctxName>"
RuleContextListDecl(rdecl) ::= "self.<rdecl.name> = list() # of <rdecl.ctxName>s"

ContextTokenGetterDecl(t)      ::= <<
def <t.name>(self):
  return self.getToken(<parser.name>.<t.name>, 0)
>>

// should never be called
ContextTokenListGetterDecl(t)  ::= <<
def <t.name>_list(self):
  return self.getTokens(<parser.name>.<t.name>)
>>

ContextTokenListIndexedGetterDecl(t)  ::= <<
def <t.name>(self, i:int=None):
  if i is None:
     return self.getTokens(<parser.name>.<t.name>)
  else:
     return self.getToken(<parser.name>.<t.name>, i)
>>

ContextRuleGetterDecl(r)       ::= <<
def <r.name>(self):
  return self.getTypedRuleContext(<parser.name>.<r.ctxName>,0)

>>

// should never be called
ContextRuleListGetterDecl(r)   ::= <<
def <r.name>_list(self):
  return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)

>>

ContextRuleListIndexedGetterDecl(r)   ::= <<
def <r.name>(self, i:int=None):
  if i is None:
     return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)
  else:
     return self.getTypedRuleContext(<parser.name>.<r.ctxName>,i)

>>

LexerRuleContext() ::= "RuleContext"
```

```
/** The rule context name is the rule followed by a suffix; e.g.,
*  r becomes rContext.
*/
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName)   ::= "_<ruleName>"
ImplicitSetLabel(id)    ::= "_tset<id>"
ListLabelName(label)    ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = self._input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = self._input.LA(1)"

StructDecl(struct,ctorAttrs,attrs,getters,dispatchMethods,interfaces,extensionMembers) ::= <<
class <struct.name>(<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif>):
  __slots__ = 'parser'

  def __init__(self, parser, parent:ParserRuleContext=None, invokingState:int=-1<struct.ctorAttrs:{a | ,
<a.name><if(a.type)>:<a.type><endif>=None}>):
    super().__init__(parent, invokingState)
    self.parser = parser
    <attrs:{a | <a>}; separator="\n">
    <struct.ctorAttrs:{a | self.<a.name> = <a.name>}; separator="\n">

  <getters:{g | <g>}; separator="\n\n">

  def getRuleIndex(self):
    return <parser.name>.RULE_<struct.derivedFromName>

<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
  def copyFrom(self, ctx:ParserRuleContext):
    super().copyFrom(ctx)
    <struct.attrs:{a | self.<a.name> = ctx.<a.name>}; separator="\n">

<endif>
  <dispatchMethods; separator="\n">
  <extensionMembers; separator="\n">

>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
class <struct.name>(<currentRule.name; format="cap">Context):

  def __init__(self, parser, ctx:ParserRuleContext): # actually a <parser.name>.<currentRule.name;
format="cap">Context
    super().__init__(parser)
    <attrs:{a | <a>}; separator="\n">
    self.copyFrom(ctx)
```

```
  <getters:{g | <g>}; separator="\n">

  <dispatchMethods; separator="\n">

>>

ListenerDispatchMethod(method) ::= <<
def <if(method.isEnter)>enter<else>exit<endif>Rule(self, listener:ParseTreeListener):
  if hasattr( listener, "<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">" ):
    listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(self)

>>

VisitorDispatchMethod(method) ::= <<
def accept(self, visitor:ParseTreeVisitor):
  if hasattr( visitor, "visit<struct.derivedFromName; format="cap">" ):
    return visitor.visit<struct.derivedFromName; format="cap">(self)
  else:
    return visitor.visitChildren(self)

>>

AttributeDecl(d) ::= "self.<d.name> = <if(d.initValue)><d.initValue><else>None<endif>"

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>localctx.<endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "localctx"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec)  ::= "self.precpred(self._ctx, <opPrec>)"
recRuleSetReturnAction(src,name)   ::= "$<name>=$<src>.<name>"
recRuleSetStopToken()                 ::= "self._ctx.stop = self._input.LT(-1)"

recRuleAltStartAction(ruleName, ctxName, label) ::= <<
localctx = <parser.name>.<ctxName>Context(self, _parentctx, _parentState)
<if(label)>localctx.<label> = _prevctx<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
localctx = <parser.name>.<currentAltLabel; format="cap">Context(self, <parser.name>.<ruleName;
format="cap">Context(self, _parentctx, _parentState))
<if(label)>
<if(isListLabel)>
localctx.<label>.append(_prevctx)
```

```
<else>
localctx.<label> = _prevctx
<endif>
<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

recRuleReplaceContext(ctxName) ::= <<
localctx = <parser.name>.<ctxName>Context(self, localctx)
self._ctx = localctx
_prevctx = localctx
>>

recRuleSetPrevCtx() ::= <<
if self._parseListeners is not None:
  self.triggerExitRuleEvent()
_prevctx = localctx
>>


LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
from antlr4 import *
from io import StringIO
import sys
if sys.version_info[1] > 5:
  from typing import TextIO
else:
  from typing.io import TextIO

<namedActions.header>

<lexer>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
<if(superClass)>
if __name__ is not None and "." in __name__:
  from .<superClass> import <superClass>
else:
  from <superClass> import <superClass>

<endif>
<atn>

class <lexer.name>(<if(superClass)><superClass><else>Lexer<endif>):

  atn = ATNDeserializer().deserialize(serializedATN())
```

```
  decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]

<if(lexer.channels)>
  <lexer.channels:{c| <c> = <lexer.channels.(c)>}; separator="\n">

<endif>
<if(rest(lexer.modes))>
  <rest(lexer.modes):{m| <m> = <i>}; separator="\n">

<endif>
  <if(lexer.tokens)>
  <lexer.tokens:{k | <k> = <lexer.tokens.(k)>}; separator="\n", wrap, anchor>
  <endif>

  channelNames = [ u"DEFAULT_TOKEN_CHANNEL", u"HIDDEN"<if (lexer.channels)>, <lexer.channels:{c|
u"<c>"}; separator=", ", wrap, anchor><endif> ]

  modeNames = [ <lexer.modes:{m| "<m>"}; separator=", ", wrap, anchor> ]

  literalNames = [ "\<INVALID>",
        <lexer.literalNames:{t | <t>}; separator=", ", wrap, anchor> ]

  symbolicNames = [ "\<INVALID>",
        <lexer.symbolicNames:{t | <t>}; separator=", ", wrap, anchor> ]

  ruleNames = [ <lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor> ]

  grammarFileName = "<lexer.grammarFileName>"

  def __init__(self, input=None, output:TextIO = sys.stdout):
     super().__init__(input, output)
     self.checkVersion("<lexerFile.ANTLRVersion>")
     self._interp = LexerATNSimulator(self, self.atn, self.decisionsToDFA, PredictionContextCache())
     self._actions = None
     self._predicates = None

  <namedActions.members>

  <dumpActions(lexer, "", actionFuncs, sempredFuncs)>

>>

SerializedATN(model) ::= <<
<! only one segment, can be inlined !>

def serializedATN():
  with StringIO() as buf:
```

```
    buf.write("<model.serialized; wrap={")<\n>       buf.write("}>")
    return buf.getvalue()

>>

/** Using a type to init value map, try to init a type; if not in table
*  must be an object, default value is "null".
*/
initValue(typeName) ::= <<
<pythonTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".py"

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/tool/templates/codegen/Python3/Python3.stg
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
```

```
/** Find left-recursive rules */
tree grammar LeftRecursiveRuleWalker;

options {
 tokenVocab=ANTLRParser;
   ASTLabelType=GrammarAST;
}

@header {
package org.antlr.v4.parse;

import org.antlr.v4.misc.*;
import org.antlr.v4.tool.*;
import org.antlr.v4.tool.ast.*;
}

@members {
private String ruleName;
private int currentOuterAltNumber; // which outer alt of rule?
public int numAlts;  // how many alts for this rule total?

public void setAltAssoc(AltAST altTree, int alt) {}
public void binaryAlt(AltAST altTree, int alt) {}
public void prefixAlt(AltAST altTree, int alt) {}
public void suffixAlt(AltAST altTree, int alt) {}
public void otherAlt(AltAST altTree, int alt) {}
public void setReturnValues(GrammarAST t) {}
}

@rulecatch { }

// TODO: can get parser errors for not matching pattern; make them go away
public
rec_rule returns [boolean isLeftRec]
@init
{
 currentOuterAltNumber = 1;
}
 : ^( r=RULE id=RULE_REF {ruleName=$id.getText();}
   ruleModifier?
//   (ARG_ACTION)? shouldn't allow args, right?
   (^(RETURNS a=ARG_ACTION {setReturnValues($a);}))?
//       ( ^(THROWS .+) )? don't allow
       ( ^(LOCALS ARG_ACTION) )? // TODO: copy these to gen'd code
       ( ^(OPTIONS .*)
       |  ^(AT ID ACTION) // TODO: copy
       )*
```

```
  ruleBlock {$isLeftRec = $ruleBlock.isLeftRec;}
  exceptionGroup
 )
;

exceptionGroup
  : exceptionHandler* finallyClause?
  ;

exceptionHandler
 : ^(CATCH ARG_ACTION ACTION)
 ;

finallyClause
 : ^(FINALLY ACTION)
 ;

ruleModifier
  : PUBLIC
  | PRIVATE
  | PROTECTED
  ;

ruleBlock returns [boolean isLeftRec]
@init{boolean lr=false; this.numAlts = $start.getChildCount();}
 : ^( BLOCK
  (
   o=outerAlternative
   {if ($o.isLeftRec) $isLeftRec = true;}
   {currentOuterAltNumber++;}
  )+
 )
;

/** An alt is either prefix, suffix, binary, or ternary operation or "other" */
outerAlternative returns [boolean isLeftRec]
  :  (binary)=>          binary
                    {binaryAlt((AltAST)$start, currentOuterAltNumber); $isLeftRec=true;}
  |  (prefix)=>          prefix
                    {prefixAlt((AltAST)$start, currentOuterAltNumber);}
  |  (suffix)=>          suffix
                    {suffixAlt((AltAST)$start, currentOuterAltNumber); $isLeftRec=true;}
  |  nonLeftRecur        {otherAlt((AltAST)$start,  currentOuterAltNumber);}
  ;

binary
 : ^( ALT elementOptions? recurse element* recurse epsilonElement* )
     {setAltAssoc((AltAST)$ALT,currentOuterAltNumber);}
```

```
;

prefix
 : ^( ALT elementOptions?
   element+
   recurse epsilonElement*
   )
       {setAltAssoc((AltAST)$ALT,currentOuterAltNumber);}
 ;

suffix
   :   ^( ALT elementOptions? recurse element+ )
       {setAltAssoc((AltAST)$ALT,currentOuterAltNumber);}
   ;

nonLeftRecur
   :   ^(ALT elementOptions? element+)
   ;

recurse
 : ^(ASSIGN ID recurseNoLabel)
 | ^(PLUS_ASSIGN ID recurseNoLabel)
 | recurseNoLabel
 ;

recurseNoLabel : {((CommonTree)input.LT(1)).getText().equals(ruleName)}? RULE_REF;

token returns [GrammarAST t=null]
 : ^(ASSIGN ID s=token {$t = $s.t;})
 | ^(PLUS_ASSIGN ID s=token {$t = $s.t;})
 | b=STRING_LITERAL         {$t = $b;}
   | ^(b=STRING_LITERAL elementOptions)  {$t = $b;}
   | ^(c=TOKEN_REF elementOptions)   {$t = $c;}
 | c=TOKEN_REF              {$t = $c;}
 ;

elementOptions
   : ^(ELEMENT_OPTIONS elementOption*)
   ;

elementOption
   : ID
   |   ^(ASSIGN ID ID)
   |   ^(ASSIGN ID STRING_LITERAL)
   |   ^(ASSIGN ID ACTION)
   |   ^(ASSIGN ID INT)
   ;
```

```
element
 : atom
 | ^(NOT element)
 | ^(RANGE atom atom)
 | ^(ASSIGN ID element)
 | ^(PLUS_ASSIGN ID element)
   | ^(SET setElement+)
   |   RULE_REF
 | ebnf
 | epsilonElement
 ;

epsilonElement
 : ACTION
 | SEMPRED
 | EPSILON
 | ^(ACTION elementOptions)
 | ^(SEMPRED elementOptions)
 ;

setElement
 : ^(STRING_LITERAL elementOptions)
 | ^(TOKEN_REF elementOptions)
 | STRING_LITERAL
 | TOKEN_REF
 ;

ebnf:   block
   |   ^( OPTIONAL block )
   |   ^( CLOSURE block )
   |   ^( POSITIVE_CLOSURE block )
   ;

block
   : ^(BLOCK ACTION? alternative+)
   ;

alternative
 : ^(ALT elementOptions? element+)
   ;

atom
 : ^(RULE_REF ARG_ACTION? elementOptions?)
   |  ^(STRING_LITERAL elementOptions)
 | STRING_LITERAL
   | ^(TOKEN_REF elementOptions)
 | TOKEN_REF
   | ^(WILDCARD elementOptions)
```

| WILDCARD
| ^(DOT ID element)
;

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/LeftRecursiveRuleWalker.g
No license file was found, but licenses were detected in source scan.

* [The "BSD license"]
* All rights reserved.
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
*   derived from this software without specific prior written permission.
[The "BSD licence"]
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
derived from this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ANTLRParser.g
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
* Copyright (c) 2012 Terence Parr
* Copyright (c) 2012 Sam Harwell
* Copyright (c) 2014 Tiago Mazzutti
* Copyright (c) 2017 Tobe Osakwe
* Copyright (c) 2020 Larry Li
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without

```
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

dartTypeInitMap ::= [
 "int":"0",
 "double":"0.0",
 "bool":"false",
 default:"null" // anything other than a primitive type is an object
]

// args must be <object-model-object>, <fields-resulting-in-STs>

ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
library <file.genPackage>;

import 'package:antlr4/antlr4.dart';
import 'dart:io';

<if(file.genListener)>
part '<file.grammarName>Listener.dart';
part '<file.grammarName>BaseListener.dart';
<endif>
<if(file.genVisitor)>
part '<file.grammarName>Visitor.dart';
part '<file.grammarName>BaseVisitor.dart';
<endif>
```

```
part '<file.grammarName>Lexer.dart';
<else>
import 'package:antlr4/antlr4.dart';
import 'dart:io';

<if(file.genListener)>
import '<file.grammarName>Listener.dart';
import '<file.grammarName>BaseListener.dart';
<endif>
<if(file.genVisitor)>
import '<file.grammarName>Visitor.dart';
import '<file.grammarName>BaseVisitor.dart';
<endif>
<endif>

<namedActions.header>
<parser>
>>

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
part of <file.genPackage>;
<else>
import 'package:antlr4/antlr4.dart';

import '<file.parserName>.dart';
<endif>
<header>

/// This abstract class defines a complete listener for a parse tree produced by
/// [<file.parserName>].
abstract class <file.grammarName>Listener extends ParseTreeListener {
<file.listenerNames:{lname |
<if(file.listenerLabelRuleNames.(lname))>
 /// Enter a parse tree produced by the [<lname>]
 /// labeled alternative in [file.parserName>.<file.listenerLabelRuleNames.(lname)>].
<else>
 /// Enter a parse tree produced by [<file.parserName>.<lname>].
<endif>
 /// [ctx] the parse tree
 void enter<lname; format="cap">(<lname; format="cap">Context ctx);
<if(file.listenerLabelRuleNames.(lname))>
 /// Exit a parse tree produced by the [<lname>]
 /// labeled alternative in [<file.parserName>.<file.listenerLabelRuleNames.(lname)>].
<else>
 /// Exit a parse tree produced by [<file.parserName>.<lname>].
<endif>
```

```
 /// [ctx] the parse tree
 void exit<lname; format="cap">(<lname; format="cap">Context ctx);}; separator="\n">
}
>>

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
part of <file.genPackage>;
<else>
import 'package:antlr4/antlr4.dart';

import '<file.parserName>.dart';
import '<file.grammarName>Listener.dart';
<endif>

<header>

/// This class provides an empty implementation of [<file.grammarName>Listener],
/// which can be extended to create a listener which only needs to handle
/// a subset of the available methods.
class <file.grammarName>BaseListener implements <file.grammarName>Listener {
<file.listenerNames:{lname |

 /// The default implementation does nothing.
 @override
 void enter<lname; format="cap">(<lname; format="cap">Context ctx) {\}

 /// The default implementation does nothing.
 @override
 void exit<lname; format="cap">(<lname; format="cap">Context ctx) {\}}; separator="\n">

 /// The default implementation does nothing.
 @override
 void enterEveryRule(ParserRuleContext ctx) {}

 /// The default implementation does nothing.
 @override
 void exitEveryRule(ParserRuleContext ctx) {}

 /// The default implementation does nothing.
 @override
 void visitTerminal(TerminalNode node) {}

 /// The default implementation does nothing.
 @override
 void visitErrorNode(ErrorNode node) {}
}
```

```
>>

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
part of <file.genPackage>;
<else>
import 'package:antlr4/antlr4.dart';

import '<file.parserName>.dart';
<endif>
<header>

/// This abstract class defines a complete generic visitor for a parse tree
/// produced by [<file.parserName>].
///
/// [T] is the eturn type of the visit operation. Use `void` for
/// operations with no return type.
abstract class <file.grammarName>Visitor\<T> extends ParseTreeVisitor\<T> {
 <file.visitorNames:{lname |
<if(file.visitorLabelRuleNames.(lname))>
/// Visit a parse tree produced by the {@code <lname>\}
/// labeled alternative in {@link <file.parserName>#<file.visitorLabelRuleNames.(lname)>\}.
<else>
/// Visit a parse tree produced by [<file.parserName>.<lname>].
<endif>
/// [ctx] the parse tree.
/// Return the visitor result.
T visit<lname; format="cap">(<lname; format="cap">Context ctx);}; separator="\n">
}
>>

BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
part of <file.genPackage>;
<else>
import 'package:antlr4/antlr4.dart';

import '<file.parserName>.dart';
import '<file.grammarName>Visitor.dart';
<endif>
<header>

/// This class provides an empty implementation of [<file.grammarName>Visitor],
/// which can be extended to create a visitor which only needs to handle
/// a subset of the available methods.
```

```
///
/// [T] is the return type of the visit operation. Use `void` for
/// operations with no return type.
class <file.grammarName>BaseVisitor\<T> extends ParseTreeVisitor\<T> implements
<file.grammarName>Visitor\<T> {
 <file.visitorNames:{lname |
/// The default implementation returns the result of calling
/// [visitChildren] on [ctx].
@override
T visit<lname; format="cap">(<lname; format="cap">Context ctx) => visitChildren(ctx);}; separator="\n">
}
>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
// Generated from <grammarFileName> by ANTLR <ANTLRVersion>
// ignore_for_file: unused_import, unused_local_variable, prefer_single_quotes
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
<if(namedActions.definitions)><namedActions.definitions><endif>
<if(parser.rules)>
const int <parser.rules:{r | RULE_<r.name> = <r.index>}; separator=", ", wrap, anchor>;
<endif>
class <parser.name> extends <superClass; null="Parser"> {
 static final checkVersion = () => RuntimeMetaData.checkVersion('<file.ANTLRVersion>',
RuntimeMetaData.VERSION);
 static const int TOKEN_EOF = IntStream.EOF;

 static final List\<DFA> _decisionToDFA = List.generate(
    _ATN.numberOfDecisions, (i) => DFA(_ATN.getDecisionState(i), i));
 static final PredictionContextCache _sharedContextCache = PredictionContextCache();
 <if(parser.tokens)>
 static const int <parser.tokens:{k | TOKEN_<k> = <parser.tokens.(k)>}; separator=", ", wrap, anchor>;
 <endif>

 @override
 final List\<String> ruleNames = [
  <parser.ruleNames:{r | '<r>'}; separator=", ", wrap, anchor>
 ];

 <vocabulary(parser.literalNames, parser.symbolicNames)>

 @override
 String get grammarFileName => '<parser.grammarFileName>';
```

```
 @override
 String get serializedATN => _serializedATN;

 @override
 ATN getATN() {
  return _ATN;
 }

 <namedActions.members>
 <parser:(ctor)()>
 <funcs; separator="\n">

<if(sempredFuncs)>
 @override
 bool sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
   switch (ruleIndex) {
   <parser.sempredFuncs.values:{f|
case <f.ruleIndex>:
 return _<f.name>_sempred(_localctx, predIndex);}; separator="\n">
   }
   return true;
 }
 <sempredFuncs.values; separator="\n">
<endif>

 <atn>
}
<funcs:{func | <if(func.ruleCtx)><func.ruleCtx><endif>}; separator="\n\n">

<funcs:{func | <if(func.altLabelCtxs)><func.altLabelCtxs:{l | <func.altLabelCtxs.(l)>};
separator="\n\n"><endif>}>
>>

vocabulary(literalNames, symbolicNames) ::= <<
static final List\<String> _LITERAL_NAMES = [
   <literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
];
static final List\<String> _SYMBOLIC_NAMES = [
   <symbolicNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
];
static final Vocabulary VOCABULARY = VocabularyImpl(_LITERAL_NAMES, _SYMBOLIC_NAMES);

@override
Vocabulary get vocabulary {
 return VOCABULARY;
}
>>
```

```
dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
void action(RuleContext _localctx, int ruleIndex, int actionIndex) {
 switch (ruleIndex) {
 <recog.actionFuncs.values:{f|
case <f.ruleIndex>:
 _<f.name>_action(_localctx, actionIndex);
 break;}; separator="\n">
 }
}
<actionFuncs.values; separator="\n">
<endif>
<if(sempredFuncs)>
bool sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
 switch (ruleIndex) {
 <recog.sempredFuncs.values:{f|
case <f.ruleIndex>:
 return _<f.name>_sempred(_localctx, predIndex);}; separator="\n">
 }
 return true;
}
<sempredFuncs.values; separator="\n">
<endif>
>>

parser_ctor(p) ::= <<
<p.name>(TokenStream input) : super(input) {
 interpreter = ParserATNSimulator(this, _ATN, _decisionToDFA, _sharedContextCache);
}
>>

/// This generates a private method since the actionIndex is generated, making an
/// overriding implementation impossible to maintain.
RuleActionFunction(r, actions) ::= <<
void _<r.name>_action(<r.ctxType> _localctx, int actionIndex) {
 switch (actionIndex) {
   <actions:{index|case <index>: <actions.(index)> break;}; separator="\n">
 }
}
>>

/// This generates a private method since the predIndex is generated, making an
/// overriding implementation impossible to maintain.
RuleSempredFunction(r, actions) ::= <<
bool _<r.name>_sempred(<r.ctxType> _localctx, int predIndex) {
 switch (predIndex) {
   <actions:{index|case <index>: return <actions.(index)>;}; separator="\n">
```

```
 }
 return true;
}
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else><endif><currentRule.ctxType>
<currentRule.name>(<args; separator=", ">) {
 dynamic _localctx = <currentRule.ctxType>(context, state<currentRule.args:{a | , <a.name>}>);
 enterRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>);
 <namedActions.init>
 <locals; separator="\n">
 try {
<if(currentRule.hasLookaheadBlock)>
  int _alt;
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
 } <if(exceptions)> <exceptions; separator="\n"><else>on RecognitionException catch (re) {
  _localctx.exception = re;
  errorHandler.reportError(this, re);
  errorHandler.recover(this, re);
 }<endif> finally {
  <finallyAction>
  exitRule();
 }
 return _localctx;
}
>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
 namedActions,finallyAction,postamble) ::=
<<

<currentRule.ctxType> <currentRule.name>([int _p = 0]<args:{a | , <a>}>) {
 final _parentctx = context;
 final _parentState = state;
 dynamic _localctx = <currentRule.ctxType>(context, _parentState<currentRule.args:{a | , <a.name>}>);
 var _prevctx = _localctx;
 var _startState = <currentRule.startState>;
 enterRecursionRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>, _p);
 <namedActions.init>
 <locals; separator="\n">
 try {
<if(currentRule.hasLookaheadBlock)>
```

```
  int _alt;
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
 } on RecognitionException catch (re) {
  _localctx.exception = re;
  errorHandler.reportError(this, re);
  errorHandler.recover(this, re);
 } finally {
  <finallyAction>
  unrollRecursionContexts(_parentctx);
 }
 return _localctx;
}
>>

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>_localctx = <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(_localctx);<endif>
enterOuterAlt(_localctx, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>

LL1AltBlock(choice, preamble, alts, error) ::= <<
state = <choice.stateNumber>;
errorHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = tokenStream.LT(1);<endif>
<preamble; separator="\n">
switch (tokenStream.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
 <alt>
 break;}; separator="\n">
default:
 <error>
}
>>

LL1OptionalBlock(choice, alts, error) ::= <<
state = <choice.stateNumber>;
errorHandler.sync(this);
switch (tokenStream.LA(1)) {
```

```
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
 <alt>
 break;}; separator="\n">
default:
 break;
}
>>

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
state = <choice.stateNumber>;
errorHandler.sync(this);
<preamble; separator="\n">
if (<expr>) {
 <alts; separator="\n">
}
<!else if ( !(<followExpr>) ) <error>!>
>>

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
state = <choice.stateNumber>;
errorHandler.sync(this);
<preamble; separator="\n">
while (<loopExpr>) {
 <alts; separator="\n">
 state = <choice.loopBackStateNumber>;
 errorHandler.sync(this);
 <iteration>
}
>>

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
state = <choice.blockStartStateNumber>; <! alt block decision !>
errorHandler.sync(this);
<preamble; separator="\n">
do {
 <alts; separator="\n">
 state = <choice.stateNumber>; <! loopback/exit decision !>
 errorHandler.sync(this);
 <iteration>
} while (<loopExpr>);
>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
state = <choice.stateNumber>;
errorHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = tokenStream.LT(1);<endif>
```

```
<preamble; separator="\n">
switch (interpreter.adaptivePredict(tokenStream, <choice.decision>, context)) {
<alts:{alt |
case <i>:
 <alt>
 break;}; separator="\n">
}
>>

OptionalBlock(choice, alts, error) ::= <<
state = <choice.stateNumber>;
errorHandler.sync(this);
switch (interpreter.adaptivePredict(tokenStream, <choice.decision>, context)) {
<alts:{alt |
case <i><if(!choice.ast.greedy)>+1<endif>:
 <alt>
 break;}; separator="\n">
}
>>

StarBlock(choice, alts, sync, iteration) ::= <<
state = <choice.stateNumber>;
errorHandler.sync(this);
_alt = interpreter.adaptivePredict(tokenStream, <choice.decision>, context);
while (_alt != <choice.exitAlt> && _alt != ATN.INVALID_ALT_NUMBER) {
 if (_alt == 1<if(!choice.ast.greedy)> + 1<endif>) {
   <iteration>
   <alts> <! should only be one !>
 }
 state = <choice.loopBackStateNumber>;
 errorHandler.sync(this);
 _alt = interpreter.adaptivePredict(tokenStream, <choice.decision>, context);
}
>>

PlusBlock(choice, alts, error) ::= <<
state = <choice.blockStartStateNumber>; <! alt block decision !>
errorHandler.sync(this);
_alt = 1<if(!choice.ast.greedy)>+1<endif>;
do {
 switch (_alt) {
 <alts:{alt|
case <i><if(!choice.ast.greedy)> + 1<endif>:
 <alt>
 break;}; separator="\n">
 default:
   <error>
 }
```

```
 state = <choice.loopBackStateNumber>; <! loopback/exit decision !>
 errorHandler.sync(this);
 _alt = interpreter.adaptivePredict(tokenStream, <choice.decision>, context);
} while (_alt != <choice.exitAlt> && _alt != ATN.INVALID_ALT_NUMBER);
>>

Sync(s) ::= "sync(<s.expecting.name>);"

ThrowNoViableAlt(t) ::= "throw NoViableAltException(this);"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShift(s.varName, bits.shift)>})> && ((BigInt.one \<\< <offsetShift(s.varName,
bits.shift)>) & (<bits.ttypes:{ttype | (BigInt.one \<\< <offsetShift({TOKEN_<ttype>}, bits.shift)>)}; separator=" |
">)) != BigInt.zero)
%>

isZero ::= [
"0":true,
default:false
]

offsetShift(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(<shiftAmount> - <offset>)<else><shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | <s.varName> == TOKEN_<ttype>}; separator=" || ">
%>

cases(ttypes) ::= <<
<ttypes:{t | case TOKEN_<t>:}; separator="\n">
>>

InvokeRule(r, argExprsChunks) ::=<<
state = <r.stateNumber>;
<if(r.labels)><r.labels:{l | <labelref(l)> =
```

```
}><endif><r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChunks>
);
>>

MatchToken(m) ::= <<
state = <m.stateNumber>;
<if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>match(TOKEN_<m.name>);
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
state = <m.stateNumber>;
<if(m.labels)><m.labels:{l | <labelref(l)> = }>tokenStream.LT(1);<endif>
<capture>
if (<if(invert)><m.varName> \<= 0 || <else>!<endif>(<expr>)) {
 <if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>errorHandler.recoverInline(this);
} else {
 if ( tokenStream.LA(1)==IntStream.EOF ) matchedEOF = true;
 errorHandler.reportMatch(this);
 consume();
}
>>

Wildcard(w) ::= <<
state = <w.stateNumber>;
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>matchWildcard();
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
state = <p.stateNumber>;
if (!(<chunks>)) {
 throw FailedPredicateException(this, <p.predicate><if(failChunks)>, <failChunks><elseif(p.msg)>,
<p.msg><endif>);
}
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
 <catchAction>
```

```
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand()  ::= "skip();"
LexerMoreCommand()  ::= "more();"
LexerPopModeCommand() ::= "popMode();"

LexerTypeCommand(arg, grammar)      ::= "type = <arg>;"
LexerChannelCommand(arg, grammar)   ::= "channel = <arg>;"
LexerModeCommand(arg, grammar)      ::= "mode_ = <arg>;"
LexerPushModeCommand(arg, grammar)  ::= "pushMode(<arg>);"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "_localctx.<a.name>"
LocalRef(a) ::= "_localctx.<a.name>"
RetValueRef(a) ::= "_localctx.<a.name>"
QRetValueRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>;"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "<ctx(t)>.<t.label>?.text"
TokenPropertyRef_type(t) ::= "<ctx(t)>.<t.label> != null ? <ctx(t)>.<t.label>.type : 0"
TokenPropertyRef_line(t) ::= "<ctx(t)>.<t.label> != null ? <ctx(t)>.<t.label>.line : 0"
TokenPropertyRef_pos(t) ::= "<ctx(t)>.<t.label> != null ? <ctx(t)>.<t.label>.charPositionInLine : 0"
TokenPropertyRef_channel(t) ::= "<ctx(t)>.<t.label> != null ? <ctx(t)>.<t.label>.channel : 0"
TokenPropertyRef_index(t) ::= "<ctx(t)>.<t.label> != null ? <ctx(t)>.<t.label>.tokenIndex : 0"
TokenPropertyRef_int(t) ::= "<ctx(t)>.<t.label> != null ? int.parse(<ctx(t)>.<t.label>.text) : 0"

RulePropertyRef_start(r) ::= "<ctx(r)>.<r.label>?.start"
RulePropertyRef_stop(r)  ::= "<ctx(r)>.<r.label>?.stop"
RulePropertyRef_text(r)  ::= "(<ctx(r)>.<r.label> != null ? tokenStream.getTextRange(<ctx(r)>.<r.label>.start,
<ctx(r)>.<r.label>.stop) : null)"
RulePropertyRef_ctx(r)  ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r)  ::= "this"

ThisRulePropertyRef_start(r) ::= "_localctx.start"
ThisRulePropertyRef_stop(r)  ::= "_localctx.stop"
ThisRulePropertyRef_text(r)  ::= "tokenStream.getTextRange(_localctx.start, tokenStream.LT(-1))"
ThisRulePropertyRef_ctx(r)  ::= "_localctx"
```

```
ThisRulePropertyRef_parser(r)  ::= "this"

NonLocalAttrRef(s)   ::= "(getInvokingContext(<s.ruleIndex>) as <s.ruleName; format=\"cap\">Context).<s.name>"
SetNonLocalAttr(s, rhsChunks)   ::=
 "(getInvokingContext(<s.ruleIndex>) as <s.ruleName; format=\"cap\">Context).<s.name> = <rhsChunks>;"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.add(<labelref(a.label)>);"

TokenDecl(t) ::= "<TokenLabelType()> <t.name>"
TokenTypeDecl(t) ::= "int <t.name>;"
TokenListDecl(t) ::= "List\<Token> <t.name> = List\<Token>()"
RuleContextDecl(r) ::= "<r.ctxName> <r.name>"
RuleContextListDecl(rdecl) ::= "List\<<rdecl.ctxName>> <rdecl.name> = List\<<rdecl.ctxName>>()"

ContextTokenGetterDecl(t) ::= <<
TerminalNode <t.name>() => getToken(<parser.name>.TOKEN_<t.name>, 0);
>>
ContextTokenListGetterDecl(t)  ::= <<
List\<TerminalNode> <t.name>s() => getTokens(<parser.name>.TOKEN_<t.name>);
>>
ContextTokenListIndexedGetterDecl(t)  ::= <<
TerminalNode <t.name>(int i) => getToken(<parser.name>.TOKEN_<t.name>, i);
>>
ContextRuleGetterDecl(r) ::= <<
<r.ctxName> <r.name>() => getRuleContext\<<r.ctxName>\>(0);
>>
ContextRuleListGetterDecl(r) ::= <<
List\<<r.ctxName>\> <r.name>s() => getRuleContexts\<<r.ctxName>\>();
>>
ContextRuleListIndexedGetterDecl(r) ::= <<
<r.ctxName> <r.name>(int i) => getRuleContext\<<r.ctxName>\>(i);
>>

LexerRuleContext() ::= "RuleContext"

/// The rule context name is the rule followed by a suffix; e.g.,
/// r becomes rContext.
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName)   ::= "_<ruleName>"
ImplicitSetLabel(id)    ::= "_tset<id>"
ListLabelName(label)    ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = tokenStream.LT(1);"
CaptureNextTokenType(d) ::= "<d.varName> = tokenStream.LA(1);"

StructDecl(struct,ctorAttrs,attrs,getters,dispatchMethods,interfaces,extensionMembers)
```

```
 ::= <<
class <struct.name> extends
<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)> implements
<interfaces; separator=", "><endif> {
 <attrs:{a | <a>;}; separator="\n">
 <getters:{g | <g>}; separator="\n">
 <struct.name>([ParserRuleContext parent, int invokingState<ctorAttrs:{a | , <a>}>]) : super(parent,
invokingState)<if(struct.ctorAttrs)> {
   <struct.ctorAttrs:{a | this.<a.name> = <a.name>;}; separator="\n">
 }<else>;<endif>

 @override
 int get ruleIndex => RULE_<struct.derivedFromName>;
<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
 @override
 void copyFrom(<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif> ctx) {
   super.copyFrom(ctx);
   <struct.attrs:{a | this.<a.name> = (ctx as <struct.name>).<a.name>;}; separator="\n">
 }
<endif>
 <dispatchMethods; separator="\n">
 <extensionMembers; separator="\n">
}
>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
class <struct.name> extends <struct.parentRule; format="cap">Context {
 <attrs:{a | <a>;}; separator="\n">
 <getters:{g | <g>}; separator="\n">
 <struct.name>(<struct.parentRule; format="cap">Context ctx) { copyFrom(ctx); }
 <dispatchMethods; separator="\n">
}
>>

ListenerDispatchMethod(method) ::= <<
@override
void <if(method.isEnter)>enter<else>exit<endif>Rule(ParseTreeListener listener) {
 if (listener is <parser.grammarName>Listener)
listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(this);
}
>>

VisitorDispatchMethod(method) ::= <<
@override
T accept\<T>(ParseTreeVisitor\<T> visitor) {
 if (visitor is <parser.grammarName>Visitor\<T>) {
  return visitor.visit<struct.derivedFromName; format="cap">(this);
 } else {
```

```
  return visitor.visitChildren(this);
 }
}
>>

AttributeDecl(d) ::= "<d.type> <d.name><if(d.initValue)> = <d.initValue><endif>"

// If we don't know location of label def x, use this template
labelref(x) ::= "<if(!x.isLocal)>_localctx.<endif><x.name>"

// For any action chunk, what is correctly-typed context struct ptr?
ctx(actionChunk) ::= "_localctx"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec)  ::= "precpred(context, <opPrec>)"
recRuleSetReturnAction(src,name)   ::= "$<name> = $<src>.<name>;"
recRuleSetStopToken()                 ::= "context.stop = tokenStream.LT(-1);"

recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<
_localctx = <ctxName>Context(_parentctx, _parentState);
<if(label)>
<if(isListLabel)>
_localctx.<label>.add(_prevctx);
<else>
_localctx.<label> = _prevctx;
<endif>
<endif>
pushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
_localctx = <currentAltLabel; format="cap">Context(new <ruleName; format="cap">Context(_parentctx,
_parentState));
<if(label)>
<if(isListLabel)>
_localctx.<label>.add(_prevctx);
<else>
_localctx.<label> = _prevctx;
<endif>
<endif>
pushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

recRuleReplaceContext(ctxName) ::= <<
_localctx = <ctxName>Context(_localctx);
context = _localctx;
_prevctx = _localctx;
>>
```

```
recRuleSetPrevCtx() ::= <<
if (parseListeners != null) triggerExitRuleEvent();
_prevctx = _localctx;
>>


LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
<if(lexerFile.genPackage)>
part of <lexerFile.genPackage>;
<else>
import 'package:antlr4/antlr4.dart';
<endif>
<namedActions.header>

<lexer>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
<if(namedActions.definitions)><namedActions.definitions><endif>

class <lexer.name> extends <superClass; null="Lexer"> {
 static final checkVersion = () => RuntimeMetaData.checkVersion('<lexerFile.ANTLRVersion>',
RuntimeMetaData.VERSION);

 static final List\<DFA> _decisionToDFA = List.generate(
      _ATN.numberOfDecisions, (i) => DFA(_ATN.getDecisionState(i), i));
 static final PredictionContextCache _sharedContextCache = PredictionContextCache();
 <if(lexer.tokens)>
 static const int
   <lexer.tokens:{k | TOKEN_<k> = <lexer.tokens.(k)>}; separator=", ", wrap, anchor>;
 <endif>
 <if(lexer.channels)>
 static const int
   <lexer.channels:{c | <c> = <lexer.channels.(c)>}; separator=", ", wrap, anchor>;
 <endif>
 <if(rest(lexer.modes))>
 static const int
   <rest(lexer.modes):{m | <m> = <i>}; separator=", ", wrap, anchor>;
 <endif>

 @override
 final List\<String> channelNames = [
   'DEFAULT_TOKEN_CHANNEL', 'HIDDEN'<if (lexer.channels)>, <lexer.channels:{c| '<c>'}; separator=", ",
wrap, anchor><endif>
 ];
```

```
@override
final List\<String> modeNames = [
  <lexer.modes:{m| '<m>'}; separator=", ", wrap, anchor>
];

@override
final List\<String> ruleNames = [
  <lexer.ruleNames:{r | '<r>'}; separator=", ", wrap, anchor>
];

<vocabulary(lexer.literalNames, lexer.symbolicNames)>

<namedActions.members>

<lexer.name>(CharStream input) : super(input) {
  interpreter = LexerATNSimulator(_ATN, _decisionToDFA, _sharedContextCache, recog: this);
}

@override
String get serializedATN => _serializedATN;

@override
String get grammarFileName => '<lexer.grammarFileName>';

@override
ATN getATN() { return _ATN; }

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>
<atn>
}
>>

SerializedATN(model) ::= <<
<if(rest(model.segments))>
<! requires segmented representation !>
static const int _serializedATNSegments = <length(model.segments)>;
<model.segments:{segment|static final String _serializedATNSegment<i0> =
 '<segment; wrap={'<\n><\t>'}>';}; separator="\n">
static final String _serializedATN = [
   <model.segments:{segment | _serializedATNSegment<i0>}; separator=",\n">
 ].join();
<else>
<! only one segment, can be inlined !>
static const String _serializedATN = '<model.serialized; wrap={'<\n><\t>'}>';
<endif>
static final ATN _ATN =
   ATNDeserializer().deserialize(_serializedATN.codeUnits);
>>
```

```
/// Using a type to init value map, try to init a type; if not in table
/// must be an object, default value is "null".
initValue(typeName) ::= <<
<dartTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".dart"
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/codegen/Dart/Dart.stg

No license file was found, but licenses were detected in source scan.

```
/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

/** How to generate rules derived from left-recursive rules.
*  These rely on recRuleAltPredicate(),
*  recRuleArg(), recRuleSetResultAction(), recRuleSetReturnAction()
*  templates in main language.stg
```

```
*/
group LeftRecursiveRules;

recRule(ruleName, argName, primaryAlts, opAlts, setResultAction,
      userRetvals, leftRecursiveRuleRefLabels) ::=
<<
<ruleName><if(userRetvals)> returns [<userRetvals>]<endif>
  :  ( {} <primaryAlts:{alt | <alt.altText> }; separator="\n       | ">
      )
      (
        <opAlts; separator="\n       | ">
      )*
  ;
>>

recRuleAlt(alt, precOption, opPrec, pred) ::= <<
{<pred>}?\<<precOption>=<opPrec>\> <alt.altText>
>>
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/LeftRecursiveRules.stg
No license file was found, but licenses were detected in source scan.


```
/*
* [The "BSD license"]
*  Copyright (c) 2015 Dan McLaughlin, Mike Lischke
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
```

* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/codegen/Cpp/Cpp.stg
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* Copyright (c) 2014 Eric Vergnaud
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

/** ANTLR tool checks output templates are compatible with tool code generation.
* For now, a simple string match used on x.y of x.y.z scheme.
* Must match Tool.VERSION during load to templates.
*
* REQUIRED.
*/

```
javascriptTypeInitMap ::= [
 "bool":"false",
 "int":"0",
 "float":"0.0",
 "str":"",
 default:"{}" // anything other than a primitive type is an object
]

// args must be <object-model-object>, <fields-resulting-in-STs>

ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
import antlr4 from 'antlr4';
<if(file.genListener)>
import <file.grammarName>Listener from './<file.grammarName>Listener.js';
<endif>
<if(file.genVisitor)>
import <file.grammarName>Visitor from './<file.grammarName>Visitor.js';
<endif>

<namedActions.header>
<parser>
>>

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
import antlr4 from 'antlr4';

// This class defines a complete listener for a parse tree produced by <file.parserName>.
export default class <file.grammarName>Listener extends antlr4.tree.ParseTreeListener {

<file.listenerNames:{lname |
 // Enter a parse tree produced by <file.parserName>#<lname>.
 enter<lname; format="cap">(ctx) {
 \}

 // Exit a parse tree produced by <file.parserName>#<lname>.
 exit<lname; format="cap">(ctx) {
 \}

}; separator="\n">

}
>>


VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
```

```
import antlr4 from 'antlr4';

// This class defines a complete generic visitor for a parse tree produced by <file.parserName>.

export default class <file.grammarName>Visitor extends antlr4.tree.ParseTreeVisitor {

<file.visitorNames:{lname |
 // Visit a parse tree produced by <file.parserName>#<lname>.
 visit<lname; format="cap">(ctx) {
   return this.visitChildren(ctx);
 \}

}; separator="\n">

}
>>


fileHeader(grammarFileName, ANTLRVersion) ::= <<
// Generated from <grammarFileName; format="java-escape"> by ANTLR <ANTLRVersion>
// jshint ignore: start
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<if(superClass)>
import <superClass> from './<superClass>.js';
<endif>

<atn>

const atn = new antlr4.atn.ATNDeserializer().deserialize(serializedATN);

const decisionsToDFA = atn.decisionToState.map( (ds, index) => new antlr4.dfa.DFA(ds, index) );

const sharedContextCache = new antlr4.PredictionContextCache();

export default class <parser.name> extends <superClass; null="antlr4.Parser"> {

   static grammarFileName = "<parser.grammarFileName; format="java-escape">";
   static literalNames = [ <parser.literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor> ];
   static symbolicNames = [ <parser.symbolicNames:{t | <t>}; null="null", separator=", ", wrap, anchor> ];
   static ruleNames = [ <parser.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor> ];

   constructor(input) {
      super(input);
      this._interp = new antlr4.atn.ParserATNSimulator(this, atn, decisionsToDFA, sharedContextCache);
      this.ruleNames = <parser.name>.ruleNames;
      this.literalNames = <parser.name>.literalNames;
```

```
    this.symbolicNames = <parser.name>.symbolicNames;
    <namedActions.members>
  }

  get atn() {
    return atn;
  }

  <if(sempredFuncs)>
  sempred(localctx, ruleIndex, predIndex) {
   switch(ruleIndex) {
   <parser.sempredFuncs.values:{f | case <f.ruleIndex>:
    return this.<f.name>_sempred(localctx, predIndex);}; separator="\n">
     default:
        throw "No predicate with index:" + ruleIndex;
    }
  }

  <sempredFuncs.values; separator="\n">
  <endif>

 <funcs; separator="\n">

}

<parser.name>.EOF = antlr4.Token.EOF;
<if(parser.tokens)>
<parser.tokens:{k | <parser.name>.<k> = <parser.tokens.(k)>;}; separator="\n", wrap, anchor>
<endif>

<if(parser.rules)>
<parser.rules:{r | <parser.name>.RULE_<r.name> = <r.index>;}; separator="\n", wrap, anchor>
<endif>

<funcs:{f | <ruleContexts(f)>}; separator="\n">

<! Define fields of this parser to export the context classes !>
<parser.funcs:{f | <parser.name>.<f.ctxType> = <f.ctxType>; }; separator="\n">

>>

ruleContexts(currentRule) ::= <<
<currentRule.ruleCtx>

<currentRule.altLabelCtxs:{l | <currentRule.altLabelCtxs.(l)>}; separator="\n">
>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
```

```
<if(actionFuncs)>
<lexer.name>.prototype.action = function(localctx, ruleIndex, actionIndex) {
 switch (ruleIndex) {
 <recog.actionFuncs.values:{f|
case <f.ruleIndex>:
 this.<f.name>_action(localctx, actionIndex);
 break;}; separator="\n">
 default:
  throw "No registered action for:" + ruleIndex;
 }
};

<actionFuncs.values; separator="\n">
<endif>
<if(sempredFuncs)>
<lexer.name>.prototype.sempred = function(localctx, ruleIndex, predIndex) {
 switch (ruleIndex) {
  <recog.sempredFuncs.values:{f| case <f.ruleIndex>:
 return this.<f.name>_sempred(localctx, predIndex);}; separator="\n">
    default:
     throw "No registered predicate for:" + ruleIndex;
   }
};

<sempredFuncs.values; separator="\n">
<endif>
>>


/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<

<lexer.name>.prototype.<r.name>_action = function(localctx , actionIndex) {
 switch (actionIndex) {
 <actions:{index|
case <index>:
 <actions.(index)>
 break;}; separator="\n">
 default:
  throw "No registered action for:" + actionIndex;
 }
};
>>

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
```

```
*/
RuleSempredFunction(r, actions) ::= <<
<if (r.factory.g.lexer)><lexer.name>.prototype.<r.name>_sempred =
function<else><r.name>_sempred<endif>(localctx, predIndex) {
 switch(predIndex) {
  <actions:{index| case <index>:
 return <actions.(index)>;}; separator="\n">
  default:
   throw "No predicate with index:" + predIndex;
 }
};

>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<


<currentRule.name>(<currentRule.args:{a | <a.name>}; separator=", ">) {
  let localctx = new <currentRule.ctxType>(this, this._ctx, this.state<currentRule.args:{a | , <a.name>}>);
  this.enterRule(localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>);
  <namedActions.init>
  <locals; separator="\n">
  try {
     <code>
     <postamble; separator="\n">
     <namedActions.after>
  }<if(exceptions)>
  <exceptions; separator="\n">
  <else> catch (re) {
   if(re instanceof antlr4.error.RecognitionException) {
       localctx.exception = re;
       this._errHandler.reportError(this, re);
       this._errHandler.recover(this, re);
    } else {
     throw re;
    }
  }<endif> finally {
     <finallyAction>
     this.exitRule();
  }
  return localctx;
}

>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
 namedActions,finallyAction,postamble) ::=
```

```
<<

<currentRule.name>(_p<if(currentRule.args)>, <args:{a | , <a>}><endif>) {
 if(_p===undefined) {
    _p = 0;
 }
  const _parentctx = this._ctx;
  const _parentState = this.state;
  let localctx = new <currentRule.ctxType>(this, this._ctx, _parentState<args:{a | , <a.name>}>);
  let _prevctx = localctx;
  const _startState = <currentRule.startState>;
  this.enterRecursionRule(localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>, _p);
  <namedActions.init>
  <locals; separator="\n">
  try {
     <code>
     <postamble; separator="\n">
     <namedActions.after>
  } catch( error) {
     if(error instanceof antlr4.error.RecognitionException) {
       localctx.exception = error;
       this._errHandler.reportError(this, error);
       this._errHandler.recover(this, error);
    } else {
     throw error;
    }
  } finally {
     <finallyAction>
     this.unrollRecursionContexts(_parentctx)
  }
  return localctx;
}

>>

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>localctx = new <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(this, localctx);<endif>
this.enterOuterAlt(localctx, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>
```

```
LL1AltBlock(choice, preamble, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = this._input.LT(1);<endif>
<preamble; separator="\n">
switch(this._input.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  break;}; separator="\n">
default:
  <error>
}
>>

LL1OptionalBlock(choice, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
switch (this._input.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
 <alt>
 break;}; separator="\n">
default:
 break;
}
>>

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<preamble; separator="\n">
if(<expr>) {
  <alts; separator="\n">
}
<!else if ( !(<followExpr>) ) <error>!>
>>

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<preamble; separator="\n">
while(<loopExpr>) {
  <alts; separator="\n">
  this.state = <choice.loopBackStateNumber>;
  this._errHandler.sync(this);
  <iteration>
}
>>
```

```
LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
this.state = <choice.blockStartStateNumber>; <! alt block decision !>
this._errHandler.sync(this);
<preamble; separator="\n">
do {
  <alts; separator="\n">
  this.state = <choice.stateNumber>; <! loopback/exit decision !>
  this._errHandler.sync(this);
  <iteration>
} while(<loopExpr>);
>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
var la_ = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx);
switch(la_) {
<alts:{alt |
case <i>:
  <alt>
  break;
}; separator="\n">
}
>>

OptionalBlock(choice, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
var la_ = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx);
<alts:{alt |
if(la_===<i><if(!choice.ast.greedy)>+1<endif>) {
  <alt>
}; separator="\n} else ">
}
>>

StarBlock(choice, alts, sync, iteration) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
var _alt = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx)
while(_alt!=<choice.exitAlt> && _alt!=antlr4.atn.ATN.INVALID_ALT_NUMBER) {
  if(_alt===1<if(!choice.ast.greedy)>+1<endif>) {
    <iteration>
    <alts> <! should only be one !>
```

```
  }
  this.state = <choice.loopBackStateNumber>;
  this._errHandler.sync(this);
  _alt = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx);
}

>>

PlusBlock(choice, alts, error) ::= <<
this.state = <choice.blockStartStateNumber>; <! alt block decision !>
this._errHandler.sync(this);
var _alt = 1<if(!choice.ast.greedy)>+1<endif>;
do {
 switch (_alt) {
 <alts:{alt|
case <i><if(!choice.ast.greedy)>+1<endif>:
 <alt>
 break;}; separator="\n">
 default:
  <error>
 }
 this.state = <choice.loopBackStateNumber>; <! loopback/exit decision !>
 this._errHandler.sync(this);
 _alt = this._interp.adaptivePredict(this._input,<choice.decision>, this._ctx);
} while ( _alt!=<choice.exitAlt> && _alt!=antlr4.atn.ATN.INVALID_ALT_NUMBER );
>>

Sync(s) ::= "sync(<s.expecting.name>)"

ThrowNoViableAlt(t) ::= "throw new antlr4.error.NoViableAltException(this);"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
>>

// Javascript language spec - shift operators are 32 bits long max
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x1f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShiftVar(s.varName, bits.shift)>})> && ((1 \<\< <offsetShiftVar(s.varName,
bits.shift)>) & (<bits.ttypes:{ttype | (1 \<\< <offsetShiftType(ttype, bits.shift)>)}; separator=" | ">)) !== 0)
%>

isZero ::= [
```

```
"0":true,
default:false
]

offsetShiftVar(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(<shiftAmount> - <offset>)<else><shiftAmount><endif>
%>

offsetShiftType(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(<parser.name>.<shiftAmount> - <offset>)<else><parser.name>.<shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | <s.varName>===<parser.name>.<ttype>}; separator=" || ">
%>

cases(ttypes) ::= <<
<ttypes:{t | case <parser.name>.<t>:}; separator="\n">
>>

InvokeRule(r, argExprsChunks) ::= <<
this.state = <r.stateNumber>;
<if(r.labels)><r.labels:{l | <labelref(l)> =
}><endif>this.<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChu
nks>);
>>

MatchToken(m) ::= <<
this.state = <m.stateNumber>;
<if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>this.match(<parser.name>.<m.name>);
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
this.state = <m.stateNumber>;
<if(m.labels)><m.labels:{l | <labelref(l)> = }>this._input.LT(1);<endif>
<capture>
<if(invert)>if(<m.varName>\<=0 || <expr>)<else>if(!(<expr>))<endif> {
  <if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>this._errHandler.recoverInline(this);
}
else {
 this._errHandler.reportMatch(this);
  this.consume();
}
```

```
>>

Wildcard(w) ::= <<
this.state = <w.stateNumber>;
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>this.matchWildcard();
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
this.state = <p.stateNumber>;
if (!( <chunks>)) {
  throw new antlr4.error.FailedPredicateException(this, <p.predicate><if(failChunks)>,
<failChunks><elseif(p.msg)>, <p.msg><endif>);
}
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
 <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand()  ::= "this.skip()"
LexerMoreCommand()  ::= "this.more()"
LexerPopModeCommand() ::= "this.popMode()"
LexerTypeCommand(arg, grammar)      ::= "this._type = <arg>"
LexerChannelCommand(arg, grammar)   ::= "this._channel = <arg>"
LexerModeCommand(arg, grammar)      ::= "this._mode = <arg>"
LexerPushModeCommand(arg, grammar)  ::= "this.pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "localctx.<a.name>"
LocalRef(a) ::= "localctx.<a.name>"
RetValueRef(a) ::= "localctx.<a.name>"
QRetValueRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"
```

```
TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(<ctx(t)>.<t.label>===null ? null : <ctx(t)>.<t.label>.text)"
TokenPropertyRef_type(t) ::= "(<ctx(t)>.<t.label> === null ? 0 : <ctx(t)>.<t.label>.type)"
TokenPropertyRef_line(t) ::= "(<ctx(t)>.<t.label> === null ? 0 : <ctx(t)>.<t.label>.line)"
TokenPropertyRef_pos(t) ::= "(<ctx(t)>.<t.label> === null ? 0 : <ctx(t)>.<t.label>.column)"
TokenPropertyRef_channel(t) ::= "(<ctx(t)>.<t.label> === null ? 0 : <ctx(t)>.<t.label>.channel)"
TokenPropertyRef_index(t) ::= "(<ctx(t)>.<t.label> === null ? 0 : <ctx(t)>.<t.label>.tokenIndex)"
TokenPropertyRef_int(t) ::= "(<ctx(t)>.<t.label> === null ? 0 : parseInt(<ctx(t)>.<t.label>.text))"

RulePropertyRef_start(r) ::= "(<ctx(r)>.<r.label>===null ? null : <ctx(r)>.<r.label>.start)"
RulePropertyRef_stop(r)  ::= "(<ctx(r)>.<r.label>===null ? null : <ctx(r)>.<r.label>.stop)"
RulePropertyRef_text(r)  ::= "(<ctx(r)>.<r.label>===null ? null : this._input.getText(new
antlr4.Interval(<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop)))"
RulePropertyRef_ctx(r)  ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r)  ::= "this"

ThisRulePropertyRef_start(r) ::= "localctx.start"
ThisRulePropertyRef_stop(r)  ::= "localctx.stop"
ThisRulePropertyRef_text(r)  ::= "this._input.getText(new antlr4.Interval(localctx.start, this._input.LT(-1)))"
ThisRulePropertyRef_ctx(r)  ::= "localctx"
ThisRulePropertyRef_parser(r)  ::= "this"

NonLocalAttrRef(s)     ::= "this.getInvokingContext(<s.ruleIndex>).<s.name>"
SetNonLocalAttr(s, rhsChunks)  ::= "this.getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.push(<labelref(a.label)>);"

TokenDecl(t) ::= "this.<t.name> = null; // <TokenLabelType()>"
TokenTypeDecl(t) ::= "var <t.name> = 0; // <TokenLabelType()> type"
TokenListDecl(t) ::= "this.<t.name> = []; // of <TokenLabelType()>s"
RuleContextDecl(r) ::= "this.<r.name> = null; // <r.ctxName>"
RuleContextListDecl(rdecl) ::= "this.<rdecl.name> = []; // of <rdecl.ctxName>s"

ContextTokenGetterDecl(t)      ::= <<
<t.name>() {
  return this.getToken(<parser.name>.<t.name>, 0);
};
>>

// should never be called
ContextTokenListGetterDecl(t)  ::= <<
<t.name>_list() {
  return this.getTokens(<parser.name>.<t.name>);
}
>>
```

```
ContextTokenListIndexedGetterDecl(t)  ::= <<
<t.name> = function(i) {
 if(i===undefined) {
  i = null;
 }
   if(i===null) {
      return this.getTokens(<parser.name>.<t.name>);
   } else {
      return this.getToken(<parser.name>.<t.name>, i);
   }
};

>>

ContextRuleGetterDecl(r)       ::= <<
<r.name>() {
   return this.getTypedRuleContext(<r.ctxName>,0);
};
>>

// should never be called
ContextRuleListGetterDecl(r)   ::= <<
<r.name>_list() {
   return this.getTypedRuleContexts(<parser.name>.<r.ctxName>);
}
>>

ContextRuleListIndexedGetterDecl(r)   ::= <<
<r.name> = function(i) {
   if(i===undefined) {
      i = null;
   }
   if(i===null) {
      return this.getTypedRuleContexts(<r.ctxName>);
   } else {
      return this.getTypedRuleContext(<r.ctxName>,i);
   }
};
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
* r becomes rContext.
*/
RuleContextNameSuffix() ::= "Context"
```

```
ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName)   ::= "_<ruleName>"
ImplicitSetLabel(id)    ::= "_tset<id>"
ListLabelName(label)    ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = self._input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = this._input.LA(1);"

StructDecl(struct,ctorAttrs,attrs,getters,dispatchMethods,interfaces,extensionMembers) ::= <<
class <struct.name> extends <if(contextSuperClass)><contextSuperClass><else>antlr4.ParserRuleContext<endif> {

  constructor(parser, parent, invokingState<struct.ctorAttrs:{a | , <a.name>}>) {
     if(parent===undefined) {
        parent = null;
     }
     if(invokingState===undefined || invokingState===null) {
        invokingState = -1;
     }
     super(parent, invokingState);
     this.parser = parser;
     this.ruleIndex = <parser.name>.RULE_<struct.derivedFromName>;
     <attrs:{a | <a>}; separator="\n">
     <struct.ctorAttrs:{a | this.<a.name> = <a.name> || null;}; separator="\n">
  }

 <getters:{g | <g>}; separator="\n\n">

 <if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
 copyFrom(ctx) {
  super.copyFrom(ctx);
  <struct.attrs:{a | this.<a.name> = ctx.<a.name>;}; separator="\n">
 }
 <endif>
 <dispatchMethods; separator="\n">
 <extensionMembers; separator="\n">

}

>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
class <struct.name> extends <struct.parentRule; format="cap">Context {

  constructor(parser, ctx) {
     super(parser);
     <attrs:{a | <a>;}; separator="\n">
     super.copyFrom(ctx);
  }
```

```
 <getters:{g | <g>}; separator="\n\n">

 <dispatchMethods; separator="\n">

}

<! Define fields of this parser to export this struct/context class !>
<parser.name>.<struct.name> = <struct.name>;

>>

ListenerDispatchMethod(method) ::= <<
<if(method.isEnter)>enter<else>exit<endif>Rule(listener) {
  if(listener instanceof <parser.grammarName>Listener ) {
     listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(this);
 }
}

>>

VisitorDispatchMethod(method) ::= <<
accept(visitor) {
  if ( visitor instanceof <parser.grammarName>Visitor ) {
     return visitor.visit<struct.derivedFromName; format="cap">(this);
  } else {
     return visitor.visitChildren(this);
  }
}

>>

AttributeDecl(d) ::= "this.<d.name> = <if(d.initValue)><d.initValue><else>null<endif>"

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>localctx.<endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "localctx"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec)  ::= "this.precpred(this._ctx, <opPrec>)"
recRuleSetReturnAction(src,name)   ::= "$<name>=$<src>.<name>"
recRuleSetStopToken()                 ::= "this._ctx.stop = this._input.LT(-1);"

recRuleAltStartAction(ruleName, ctxName, label) ::= <<
localctx = new <ctxName>Context(this, _parentctx, _parentState);
<if(label)>localctx.<label> = _prevctx;<endif>
```

```
this.pushNewRecursionContext(localctx, _startState, <parser.name>.RULE_<ruleName>);
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
localctx = new <currentAltLabel; format="cap">Context(this, new <ruleName; format="cap">Context(this,
_parentctx, _parentState));
<if(label)>
<if(isListLabel)>
localctx.<label>.push(_prevctx);
<else>
localctx.<label> = _prevctx;
<endif>
<endif>
this.pushNewRecursionContext(localctx, _startState, <parser.name>.RULE_<ruleName>);
>>

recRuleReplaceContext(ctxName) ::= <<
localctx = new <ctxName>Context(this, localctx);
this._ctx = localctx;
_prevctx = localctx;
>>

recRuleSetPrevCtx() ::= <<
if(this._parseListeners!==null) {
  this.triggerExitRuleEvent();
}
_prevctx = localctx;
>>


LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
import antlr4 from 'antlr4';

<namedActions.header>

<lexer>

>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
<if(superClass)>
import <superClass> from './<superClass>.js';
<endif>

<atn>

const atn = new antlr4.atn.ATNDeserializer().deserialize(serializedATN);
```

```
const decisionsToDFA = atn.decisionToState.map( (ds, index) => new antlr4.dfa.DFA(ds, index) );

export default class <lexer.name> extends <if(superClass)><superClass><else>antlr4.Lexer<endif> {

  static grammarFileName = "<lexer.grammarFileName>";
  static channelNames = [ "DEFAULT_TOKEN_CHANNEL", "HIDDEN"<if (lexer.channels)>,
<lexer.channels:{c| "<c>"}; separator=", ", wrap, anchor><endif> ];
 static modeNames = [ <lexer.modes:{m| "<m>"}; separator=", ", wrap, anchor> ];
 static literalNames = [ <lexer.literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor> ];
 static symbolicNames = [ <lexer.symbolicNames:{t | <t>}; null="null", separator=", ", wrap, anchor> ];
 static ruleNames = [ <lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor> ];

  constructor(input) {
     super(input)
     this._interp = new antlr4.atn.LexerATNSimulator(this, atn, decisionsToDFA, new
antlr4.PredictionContextCache());
     <namedActions.members>
  }

  get atn() {
     return atn;
  }
}

<lexer.name>.EOF = antlr4.Token.EOF;
<if(lexer.tokens)>
<lexer.tokens:{k | <lexer.name>.<k> = <lexer.tokens.(k)>;}; separator="\n", wrap, anchor>
<endif>

<if(lexer.channels)>
<lexer.channels:{c| <lexer.name>.<c> = <lexer.channels.(c)>;}; separator="\n">

<endif>
<if(rest(lexer.modes))>
<rest(lexer.modes):{m| <lexer.name>.<m> = <i>;}; separator="\n">

<endif>

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>


>>

SerializedATN(model) ::= <<
<! only one segment, can be inlined !>

const serializedATN = ["<model.serialized; wrap={",<\n>    "}>"].join("");
```

```
>>

/** Using a type to init value map, try to init a type; if not in table
* must be an object, default value is "null".
*/
initValue(typeName) ::= <<
<javacriptTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".js"
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/codegen/JavaScript/JavaScript.stg
No license file was found, but licenses were detected in source scan.


```
/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  Copyright (c) 2015 Janyou
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
```

```
SwiftTypeInitMap ::= [
 "Int":"0",
 "Int64":"0",
 "Float":"0.0",
 "Double":"0.0",
 "Bool":"false",
 default:"nil" // anything other than a primitive type is an object
]
SwiftTypeMap ::= [
 "int":"Int",
 "float":"Float",
 "long":"Int64",
 "double":"Double",
 "bool":"Bool",
 "boolean":"Bool",
 default : key
]
// args must be <object-model-object>, <fields-resulting-in-STs>

accessLevelOpenOK(obj) ::= "<obj.accessLevel; null=\"open\">"
accessLevelNotOpen(obj) ::= "<obj.accessLevel; null=\"public\">"

ParserFile(file, parser, namedActions,contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<namedActions.header>
import Antlr4

<parser>
>>

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>
import Antlr4

/**
* This interface defines a complete listener for a parse tree produced by
* {@link <file.parserName>}.
*/
<accessLevelNotOpen(file)> protocol <file.grammarName>Listener: ParseTreeListener {
 <file.listenerNames:{lname |
/**
```

```
<if(file.listenerLabelRuleNames.(lname))>
* Enter a parse tree produced by the {@code <lname>\}
* labeled alternative in {@link <file.parserName>#<file.listenerLabelRuleNames.(lname)>\}.
<else>
* Enter a parse tree produced by {@link <file.parserName>#<lname>\}.
<endif>
- Parameters:
  - ctx: the parse tree
*/
func enter<lname; format="cap">(_ ctx: <file.parserName>.<lname; format="cap">Context)
/**
<if(file.listenerLabelRuleNames.(lname))>
* Exit a parse tree produced by the {@code <lname>\}
* labeled alternative in {@link <file.parserName>#<file.listenerLabelRuleNames.(lname)>\}.
<else>
* Exit a parse tree produced by {@link <file.parserName>#<lname>\}.
<endif>
- Parameters:
  - ctx: the parse tree
*/
func exit<lname; format="cap">(_ ctx: <file.parserName>.<lname; format="cap">Context)}; separator="\n">
}
>>

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>

import Antlr4


/**
* This class provides an empty implementation of {@link <file.grammarName>Listener},
* which can be extended to create a listener which only needs to handle a subset
* of the available methods.
*/
<accessLevelOpenOK(file)> class <file.grammarName>BaseListener: <file.grammarName>Listener {
    <accessLevelNotOpen(file)> init() { \}
 <file.listenerNames:{lname |

/**
* {@inheritDoc\}
*
* \<p>The default implementation does nothing.\</p>
*/
```

```
<accessLevelOpenOK(file)> func enter<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) { \}
/**
* {@inheritDoc\}
*
* \<p>The default implementation does nothing.\</p>
*/
<accessLevelOpenOK(file)> func exit<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) { \}}; separator="\n">

 /**
  * {@inheritDoc\}
  *
  * \<p>The default implementation does nothing.\</p>
  */
 <accessLevelOpenOK(file)> func enterEveryRule(_ ctx: ParserRuleContext) throws { }
 /**
  * {@inheritDoc\}
  *
  * \<p>The default implementation does nothing.\</p>
  */
 <accessLevelOpenOK(file)> func exitEveryRule(_ ctx: ParserRuleContext) throws { }
 /**
  * {@inheritDoc\}
  *
  * \<p>The default implementation does nothing.\</p>
  */
 <accessLevelOpenOK(file)> func visitTerminal(_ node: TerminalNode) { }
 /**
  * {@inheritDoc\}
  *
  * \<p>The default implementation does nothing.\</p>
  */
 <accessLevelOpenOK(file)> func visitErrorNode(_ node: ErrorNode) { }
}
>>

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>
import Antlr4

/**
* This interface defines a complete generic visitor for a parse tree produced
* by {@link <file.parserName>}.
```

```
*
* @param \<T> The return type of the visit operation. Use {@link Void} for
* operations with no return type.
*/
<accessLevelOpenOK(file)> class <file.grammarName>Visitor\<T>: ParseTreeVisitor\<T> {
 <file.visitorNames:{lname |
/**
<if(file.visitorLabelRuleNames.(lname))>
* Visit a parse tree produced by the {@code <lname>\}
* labeled alternative in {@link <file.parserName>#<file.visitorLabelRuleNames.(lname)>\}.
<else>
* Visit a parse tree produced by {@link <file.parserName>#<lname>\}.
<endif>
- Parameters:
 - ctx: the parse tree
- returns: the visitor result
*/
<accessLevelOpenOK(file)> func visit<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) -> T {
 fatalError(#function + " must be overridden")
\}
}; separator="\n">
}
>>

BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>
import Antlr4

/**
* This class provides an empty implementation of {@link <file.grammarName>Visitor},
* which can be extended to create a visitor which only needs to handle a subset
* of the available methods.
*
* @param \<T> The return type of the visit operation. Use {@link Void} for
* operations with no return type.
*/
<accessLevelOpenOK(file)> class <file.grammarName>BaseVisitor\<T>: AbstractParseTreeVisitor\<T> {
 <file.visitorNames:{lname |
/**
* {@inheritDoc\}
*
* \<p>The default implementation returns the result of calling
* {@link #visitChildren\} on {@code ctx\}.\</p>
```

```
*/
<accessLevelOpenOK(file)> func visit<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) -> T? { return visitChildren(ctx) \}}; separator="\n">
}
>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
// Generated from <grammarFileName; format="java-escape"> by ANTLR <ANTLRVersion>
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
<!//@SuppressWarnings({"all", "warnings", "unchecked", "unused", "cast"})!>
<accessLevelOpenOK(parser)> class <parser.name>: <superClass; null="Parser"> {

 internal static var _decisionToDFA: [DFA] = {
        var decisionToDFA = [DFA]()
        let length = <parser.name>._ATN.getNumberOfDecisions()
        for i in 0..\<length {
         <!// decisionToDFA[i] = DFA(<parser.name>._ATN.getDecisionState(i)!, i);!>
          decisionToDFA.append(DFA(<parser.name>._ATN.getDecisionState(i)!, i))
         }
         return decisionToDFA
    }()

 internal static let _sharedContextCache = PredictionContextCache()

 <if(parser.tokens)>
 <accessLevelNotOpen(parser)>
 enum Tokens: Int {
  case EOF = -1, <parser.tokens:{k | <k> = <parser.tokens.(k)>}; separator=", ", wrap, anchor>
 }
 <endif>

 <accessLevelNotOpen(parser)>
 <if(parser.rules)>
 static let <parser.rules:{r | RULE_<r.name> = <r.index>}; separator=", ", wrap, anchor>
 <endif>

 <accessLevelNotOpen(parser)>
 static let ruleNames: [String] = [
  <parser.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
 ]

 <vocabulary(parser.literalNames, parser.symbolicNames,
```

```
                accessLevelNotOpen(parser))>

 override <accessLevelOpenOK(parser)>
 func getGrammarFileName() -> String { return "<parser.grammarFileName; format="java-escape">" }

 override <accessLevelOpenOK(parser)>
 func getRuleNames() -> [String] { return <parser.name>.ruleNames }

 override <accessLevelOpenOK(parser)>
 func getSerializedATN() -> String { return <parser.name>._serializedATN }

 override <accessLevelOpenOK(parser)>
 func getATN() -> ATN { return <parser.name>._ATN }

 <namedActions.members>
 <parser:(ctor)()>
 <funcs; separator="\n">

<if(sempredFuncs)>
 override <accessLevelOpenOK(parser)>
 func sempred(_ _localctx: RuleContext?, _ ruleIndex: Int,  _ predIndex: Int)throws -> Bool {
  switch (ruleIndex) {
  <parser.sempredFuncs.values:{f|
case  <f.ruleIndex>:
 return try <f.name>_sempred(_localctx?.castdown(<f.ctxType>.self), predIndex)}; separator="\n">
     default: return true
  }
  <!return true;!>
 }
 <sempredFuncs.values; separator="\n">
<endif>

 <atn>

 <accessLevelNotOpen(parser)>
 static let _serializedATN = <parser.name>ATN().jsonString

 <accessLevelNotOpen(parser)>
 static let _ATN = ATNDeserializer().deserializeFromJson(_serializedATN)
}
>>

vocabulary(literalNames, symbolicNames, accessLevel) ::= <<
private static let _LITERAL_NAMES: [String?] = [
 <literalNames:{t | <t>}; null="nil", separator=", ", wrap, anchor>
]
private static let _SYMBOLIC_NAMES: [String?] = [
 <symbolicNames:{t | <t>}; null="nil", separator=", ", wrap, anchor>
```

```
]
<accessLevel>
static let VOCABULARY = Vocabulary(_LITERAL_NAMES, _SYMBOLIC_NAMES)
>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
override <accessLevelOpenOK(parser)>
func action(_ _localctx: RuleContext?,  _ ruleIndex: Int, _ actionIndex: Int) throws {
 switch (ruleIndex) {
 <recog.actionFuncs.values:{f|
case <f.ruleIndex>:
 <f.name>_action((_localctx as <f.ctxType>?), actionIndex)
  }; separator="\n">
 default: break
 }
}
<actionFuncs.values; separator="\n">
<endif>
<if(sempredFuncs)>
override <accessLevelOpenOK(parser)>
func sempred(_ _localctx: RuleContext?, _  ruleIndex: Int,_   predIndex: Int) throws -> Bool {
 switch (ruleIndex) {
 <recog.sempredFuncs.values:{f|
case <f.ruleIndex>:
 return try <f.name>_sempred(_localctx?.castdown(<f.ctxType>.self), predIndex)}; separator="\n">
 default: return true
 }
 <!return true;!>
}
<sempredFuncs.values; separator="\n">
<endif>
>>

parser_ctor(p) ::= <<

override <accessLevelOpenOK(parser)>
func getVocabulary() -> Vocabulary {
   return <p.name>.VOCABULARY
}

override <accessLevelNotOpen(parser)>
init(_ input:TokenStream) throws {
   RuntimeMetaData.checkVersion("4.9.2", RuntimeMetaData.VERSION)
 try super.init(input)
 _interp = ParserATNSimulator(self,<p.name>._ATN,<p.name>._decisionToDFA,
<parser.name>._sharedContextCache)
}
```

```
>>

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<
private func <r.name>_action(_ _localctx: <r.ctxType>?,  _ actionIndex: Int) {
 switch (actionIndex) {
 <actions:{index|
case <index>:
 <actions.(index)>
  }; separator="\n">
  default: break
 }
}
>>

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
private func <r.name>_sempred(_ _localctx: <r.ctxType>!,  _ predIndex: Int) throws -> Bool {
 switch (predIndex) {
 <actions:{index|
   case <index>:return <actions.(index)>}; separator="\n">
     default: return true
 }
 <!return true;!>
}
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<
<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">
@discardableResult
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else> <accessLevelOpenOK(parser)> func
<endif><currentRule.name>(<if(first(args))>_ <endif><args; separator=", _">) throws -> <currentRule.ctxType> {
 var _localctx: <currentRule.ctxType> = <currentRule.ctxType>(_ctx, getState()<currentRule.args:{a | ,
<a.name>}>)
 try enterRule(_localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>)
 <namedActions.init>
 <locals; separator="\n">
 defer {
    <finallyAction>
    try! exitRule()
  }
```

```
 do {
<if(currentRule.hasLookaheadBlock)>
  var _alt:Int
<endif>
   <code>
  <postamble; separator="\n">
   <namedActions.after>
 }
 <if(exceptions)>
 <exceptions; separator="\n">
 <else>
 catch ANTLRException.recognition(let re) {
  _localctx.exception = re
  _errHandler.reportError(self, re)
  try _errHandler.recover(self, re)
 }
 <endif>

 return _localctx
}
>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
 namedActions,finallyAction,postamble) ::=
<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else> <accessLevelNotOpen(parser)> final <endif>
func <currentRule.name>( <if(first(args))>_ <endif><args; separator=", _">) throws -> <currentRule.ctxType>    {
 return try <currentRule.name>(0<currentRule.args:{a | , <a.name>}>)
}
@discardableResult
private func <currentRule.name>(_ _p<args:{a | , <a>}>: Int) throws -> <currentRule.ctxType>    {
 let _parentctx: ParserRuleContext? = _ctx
 let _parentState: Int = getState()
 var _localctx: <currentRule.ctxType> = <currentRule.ctxType>(_ctx, _parentState<currentRule.args:{a | ,
<a.name>}>)
 var _prevctx: <currentRule.ctxType> = _localctx
 let _startState: Int = <currentRule.startState>
 try enterRecursionRule(_localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>, _p)
 <namedActions.init>
 <locals; separator="\n">
 defer {
    <finallyAction>
    try! unrollRecursionContexts(_parentctx)
  }
```

```
 do {
<if(currentRule.hasLookaheadBlock)>
  var _alt: Int
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
 }
 catch ANTLRException.recognition(let re) {
  _localctx.exception = re
  _errHandler.reportError(self, re)
  try _errHandler.recover(self, re)
 }

 return _localctx;
}
>>

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>_localctx =  <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(_localctx);<endif>
try enterOuterAlt(_localctx, <currentOuterMostAltCodeBlock.alt.altNum>)
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<!//{!>
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
<!//}!>
>>

LL1AltBlock(choice, preamble, alts, error) ::= <<
setState(<choice.stateNumber>)
try _errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = try _input.LT(1)<endif>
<preamble; separator="\n">
switch (<parser.name>.Tokens(rawValue: try _input.LA(1))!) {
<choice.altLook,alts:{look,alt | <cases(ttypes=look)>
 <alt>
 break}; separator="\n">
default:
 <error>
}
>>

LL1OptionalBlock(choice, alts, error) ::= <<
```

```
setState(<choice.stateNumber>)
try _errHandler.sync(self)
switch (<parser.name>.Tokens(rawValue: try _input.LA(1))!) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
 <alt>
 break}; separator="\n">
default:
 break
}
>>

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
setState(<choice.stateNumber>)
try _errHandler.sync(self)
<preamble; separator="\n">
if (<expr>) {
 <alts; separator="\n">
}
<!else if ( !(<followExpr>) ) <error>!>
>>

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
setState(<choice.stateNumber>)
try _errHandler.sync(self)
<preamble; separator="\n">
while (<loopExpr>) {
 <alts; separator="\n">
 setState(<choice.loopBackStateNumber>)
 try _errHandler.sync(self)
 <iteration>
}
>>

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
setState(<choice.blockStartStateNumber>) <! alt block decision !>
try _errHandler.sync(self)
<preamble; separator="\n">
repeat {
 <alts; separator="\n">
 setState(<choice.stateNumber>); <! loopback/exit decision !>
 try _errHandler.sync(self)
 <iteration>
} while (<loopExpr>)
>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
```

```
setState(<choice.stateNumber>)
try _errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = try _input.LT(1)<endif>
<preamble; separator="\n">
switch(try getInterpreter().adaptivePredict(_input,<choice.decision>, _ctx)) {
<alts:{alt |
case <i>:
 <alt>
 break}; separator="\n">
default: break
}
>>

OptionalBlock(choice, alts, error) ::= <<
setState(<choice.stateNumber>)
try _errHandler.sync(self)
switch (try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)) {
<alts:{alt |
case <i><if(!choice.ast.greedy)>+1<endif>:
 <alt>
 break}; separator="\n">
default: break
}
>>

StarBlock(choice, alts, sync, iteration) ::= <<
setState(<choice.stateNumber>)
try _errHandler.sync(self)
_alt = try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)
while (_alt != <choice.exitAlt> && _alt != ATN.INVALID_ALT_NUMBER) {
 if ( _alt==1<if(!choice.ast.greedy)>+1<endif> ) {
  <iteration>
  <alts> <! should only be one !>
 }
 setState(<choice.loopBackStateNumber>)
 try _errHandler.sync(self)
 _alt = try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)
}
>>

PlusBlock(choice, alts, error) ::= <<
setState(<choice.blockStartStateNumber>); <! alt block decision !>
try _errHandler.sync(self)
_alt = 1<if(!choice.ast.greedy)>+1<endif>;
repeat {
 switch (_alt) {
 <alts:{alt|
case <i><if(!choice.ast.greedy)>+1<endif>:
```

```
 <alt>
 break}; separator="\n">
 default:
  <error>
 }
 setState(<choice.loopBackStateNumber>); <! loopback/exit decision !>
 try _errHandler.sync(self)
 _alt = try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)
} while (_alt != <choice.exitAlt> && _alt !=  ATN.INVALID_ALT_NUMBER)
>>

Sync(s) ::= "sync(<s.expecting.name>);"

ThrowNoViableAlt(t) ::= "throw ANTLRException.recognition(e: NoViableAltException(self))"

TestSetInline(s) ::= <<
<!<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">!>
//closure
{ () -> Bool in
    <if(rest(s.bitsets))>var<else>let<endif> testSet: Bool = <first(s.bitsets):{bits |
<if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s, bits)><endif>}>
       <rest(s.bitsets):{bits | testSet = testSet || <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s,
bits)><else><bitsetInlineComparison(s, bits)><endif>}; separator="\n">
    return testSet
}()
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <<
<!(<testShiftInRange({<offsetShift(s.varName, bits.shift)>})> && ((1 \<\< <offsetShift(s.varName, bits.shift)>) &
(<bits.ttypes:{ttype | (1 \<\< <offsetShift(ttype, bits.shift)>)}; separator=" | ">)) != 0)!>
{  () -> Bool in
  <! let test: Bool = (<testShiftInRange({<offsetShift(s.varName, bits.shift)>})>)!>
  <!var temp: Int64 =  Int64(<offsetShift(s.varName, bits.shift)>)!>
  <!temp = (temp \< 0) ? (64 + (temp % 64 )) : (temp % 64)!>
  <!let test1: Int64 = (Int64(1) \<\< temp)!>
  <!var test2: Int64 = (<first(bits.ttypes):{ttype | Utils.bitLeftShift(<offsetShift(parserName(ttype), bits.shift)>)}>)!>
  <!<rest(bits.ttypes):{ttype | test2 = test2 | Utils.bitLeftShift(<offsetShift(parserName(ttype), bits.shift)>)};
separator="\n">!>
  let testArray: [Int] = [<s.varName>, <bits.ttypes:{ttype |<parserName(ttype)>}; separator=",">]
  <!var test2: Int64 = Utils.testBitLeftShiftArray(testArray)!>
   return  Utils.testBitLeftShiftArray(testArray, <bits.shift>)
```

```
}()
>>

isZero ::= [
"0": true,
default: false
]
parserName(ttype) ::= <%
<parser.name>.Tokens.<ttype>.rawValue
%>
offsetShift(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(<shiftAmount> - <offset>)<else><shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | <s.varName> == <parser.name>.Tokens.<ttype>.rawValue}; separator=" || ">
%>

cases(ttypes) ::= <<
<trunc(ttypes): {t | case .<t>:fallthrough}  ; separator="\n">
<last(ttypes): {t | case .<t>:}  ; separator="\n">
>>

InvokeRule(r, argExprsChunks) ::= <<
setState(<r.stateNumber>)
<if(r.labels)>
try {
  let assignmentValue = try
<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChunks>)
  <r.labels:{l | <labelref(l)> = assignmentValue} ; separator="\n">
    }()
<else>try
<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChunks>)<endif>
>>

MatchToken(m) ::= <<
setState(<m.stateNumber>)
<if(m.labels)>
try {
  let assignmentValue = try match(<parser.name>.Tokens.<m.name>.rawValue)
  <m.labels:{l | <labelref(l)> = assignmentValue}  ; separator="\n">
    }()
<else>try match(<parser.name>.Tokens.<m.name>.rawValue)<endif>
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"
```

```
MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
setState(<m.stateNumber>)
<if(m.labels)><m.labels:{l | <labelref(l)> = }>try _input.LT(1)<endif>
<capture>
if (<if(invert)><m.varName> \<= 0 || <else>!<endif>(<expr>)) {
 <if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>try _errHandler.recoverInline(self)<if(m.labels)> as
Token<endif>
}
else {
 _errHandler.reportMatch(self)
 try consume()
}
>>

Wildcard(w) ::= <<
setState(<w.stateNumber>)
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>try matchWildcard();
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
setState(<p.stateNumber>)
if (!(<chunks>)) {
  throw ANTLRException.recognition(e:FailedPredicateException(self, <p.predicate><if(failChunks)>,
<failChunks><elseif(p.msg)>, <p.msg><endif>))
}
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
 <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand()  ::= "skip()"
LexerMoreCommand()  ::= "more()"
LexerPopModeCommand() ::= "popMode()"

LexerTypeCommand(arg)      ::= "_type = <arg>"
```

```
LexerChannelCommand(arg)   ::= "_channel = <arg>"
LexerModeCommand(arg)      ::= "_mode = <arg>"
LexerPushModeCommand(arg)  ::= "pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "_localctx.<a.name>"
LocalRef(a) ::= "_localctx.<a.name>"
RetValueRef(a) ::= "_localctx.<a.name>"
QRetValueRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getText()! : \"\")"
TokenPropertyRef_type(t) ::= "(<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getType() : 0)"
TokenPropertyRef_line(t) ::= "(<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getLine() : 0)"
TokenPropertyRef_pos(t) ::= "(<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getCharPositionInLine() : 0)"
TokenPropertyRef_channel(t) ::= "(<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getChannel() : 0)"
TokenPropertyRef_index(t) ::= "(<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getTokenIndex() : 0)"
TokenPropertyRef_int(t) ::= "(<ctx(t)>.<t.label> != nil ? Int(<ctx(t)>.<t.label>!.getText()!) : 0)"

RulePropertyRef_start(r) ::= "(<ctx(r)>.<r.label> != nil ? (<ctx(r)>.<r.label>!.start?.description ?? \"\") : \"\")"
RulePropertyRef_stop(r)  ::= "(<ctx(r)>.<r.label> != nil ? (<ctx(r)>.<r.label>!.stop?.description ?? \"\") : \"\")"
RulePropertyRef_text(r)  ::= "(<ctx(r)>.<r.label> != nil ? try
_input.getText(<ctx(r)>.<r.label>!.start,<ctx(r)>.<r.label>!.stop) : \"\")"
RulePropertyRef_ctx(r)  ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r)  ::= "self"

ThisRulePropertyRef_start(r) ::= "_localctx.start"
ThisRulePropertyRef_stop(r)  ::= "_localctx.stop"
ThisRulePropertyRef_text(r)  ::= "(try _input.getText(_localctx.start, try _input.LT(-1)))"
ThisRulePropertyRef_ctx(r)  ::= "_localctx"
ThisRulePropertyRef_parser(r)  ::= "self"

NonLocalAttrRef(s)   ::= "((<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>)).<s.name>"
SetNonLocalAttr(s, rhsChunks)   ::=
 "((<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>)).<s.name> = <rhsChunks>"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.append(<labelref(a.label)>)"

TokenDecl(t) ::= "<t.name>: <SwiftTypeMap.(TokenLabelType())>!"
TokenTypeDecl(t) ::= "var <t.name>: Int = 0"
```

```
TokenListDecl(t) ::= "<t.name>: [Token] = [Token]()"
RuleContextDecl(r) ::= "<r.name>: <r.ctxName>!"
RuleContextListDecl(rdecl) ::= "<rdecl.name>: [<rdecl.ctxName>] = [<rdecl.ctxName>]()"

ContextTokenGetterDecl(t) ::= <<
 <accessLevelOpenOK(parser)>
 func <t.name>() -> TerminalNode? {
  return getToken(<parser.name>.Tokens.<t.name>.rawValue, 0)
 }
>>
ContextTokenListGetterDecl(t) ::= <<
 <accessLevelOpenOK(parser)>
 func <t.name>() -> [TerminalNode] {
  return getTokens(<parser.name>.Tokens.<t.name>.rawValue)
 }
>>
ContextTokenListIndexedGetterDecl(t) ::= <<
 <accessLevelOpenOK(parser)>
 func <t.name>(_ i:Int) -> TerminalNode? {
  return getToken(<parser.name>.Tokens.<t.name>.rawValue, i)
 }
>>
ContextRuleGetterDecl(r) ::= <<
 <accessLevelOpenOK(parser)>
 func <r.name>() -> <r.ctxName>? {
  return getRuleContext(<r.ctxName>.self, 0)
 }
>>
ContextRuleListGetterDecl(r) ::= <<
 <accessLevelOpenOK(parser)>
 func <r.name>() -> [<r.ctxName>] {
  return getRuleContexts(<r.ctxName>.self)
 }
>>
ContextRuleListIndexedGetterDecl(r)   ::= <<
 <accessLevelOpenOK(parser)>
 func <r.name>(_ i: Int) -> <r.ctxName>? {
  return getRuleContext(<r.ctxName>.self, i)
 }
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
* r becomes rContext.
*/
RuleContextNameSuffix() ::= "Context"
```

```
ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName)   ::= "_<ruleName>"
ImplicitSetLabel(id)    ::= "_tset<id>"
ListLabelName(label)    ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = try _input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = try _input.LA(1)"

StructDecl(struct,ctorAttrs,attrs,getters,dispatchMethods,interfaces,extensionMembers,
        superClass={ParserRuleContext}) ::= <<

<accessLevelNotOpen(parser)> class <struct.name>:
<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)>, <interfaces;
separator=", "><endif> {
 <attrs:{a | <accessLevelOpenOK(parser)> var <a>}; separator="\n">
 <getters:{g | <g>}; separator="\n">
 <! <if(ctorAttrs)> <accessLevelNotOpen(parser)> init(_ parent: ParserRuleContext,_ invokingState: Int) {
super.init(parent, invokingState)  }<endif> !>
 <if(ctorAttrs)>
 <accessLevelNotOpen(parser)> convenience init(_ parent: ParserRuleContext?, _ invokingState: Int<ctorAttrs:{a | ,
_ <a>}>) {
  self.init(parent, invokingState)
  <struct.ctorAttrs:{a | self.<a.name> = <a.name>;}; separator="\n">
 }
 <endif>

 override <accessLevelOpenOK(parser)>
 func getRuleIndex() -> Int {
  return <parser.name>.RULE_<struct.derivedFromName>
 }
<if(struct.provideCopyFrom && struct.attrs)> <! don't need copy unless we have subclasses !>
 <accessLevelOpenOK(parser)>
 override func copyFrom(_ ctx_: ParserRuleContext) {
  super.copyFrom(ctx_)
  let ctx = ctx_ as! <struct.name>
  <struct.attrs:{a | self.<a.name> = ctx.<a.name>;}; separator="\n">
 }
<endif>
 <dispatchMethods; separator="\n">
 <extensionMembers; separator="\n">
}
>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
<accessLevelNotOpen(parser)> class <struct.name>: <currentRule.name; format="cap">Context {
 <attrs:{a | <accessLevelNotOpen(parser)> var <a>}; separator="\n">
 <getters:{g | <g>}; separator="\n">
```

```
 <accessLevelNotOpen(parser)>
 init(_ ctx: <currentRule.name; format="cap">Context) {
  super.init()
  copyFrom(ctx)
 }
 <dispatchMethods; separator="\n">
}
>>

ListenerDispatchMethod(method) ::= <<
override <accessLevelOpenOK(parser)>
func <if(method.isEnter)>enter<else>exit<endif>Rule(_ listener: ParseTreeListener) {
 if let listener = listener as? <parser.grammarName>Listener {
  listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(self)
 }
}
>>

VisitorDispatchMethod(method) ::= <<
override <accessLevelOpenOK(parser)>
func accept\<T>(_ visitor: ParseTreeVisitor\<T>) -> T? {
 if let visitor = visitor as? <parser.grammarName>Visitor {
    return visitor.visit<struct.derivedFromName; format="cap">(self)
 }
 else if let visitor = visitor as? <parser.grammarName>BaseVisitor {
    return visitor.visit<struct.derivedFromName; format="cap">(self)
 }
 else {
     return visitor.visitChildren(self)
 }
}
>>

AttributeDecl(d) ::= "<d.name>: <SwiftTypeMap.(d.type)><if(d.initValue)> = <d.initValue><else>!<endif>"

/** If we don't know location of label def x, use this template (_localctx as! <x.ctx.name> */
labelref(x) ::= "<if(!x.isLocal)>_localctx.castdown(<x.ctx.name>.self).<endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr?    */
ctx(actionChunk) ::= "_localctx.castdown(<actionChunk.ctx.name>.self)"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec)  ::= "precpred(_ctx, <opPrec>)"
recRuleSetReturnAction(src,name)   ::= "$<name>=$<src>.<name>"
recRuleSetStopToken()                 ::= "_ctx!.stop = try _input.LT(-1)"

recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<
_localctx = <ctxName>Context(_parentctx, _parentState);
```

```
<if(label)>
<if(isListLabel)>
_localctx.<label>.append(_prevctx)
<else>
_localctx.<label> = _prevctx
<endif>
<endif>
<if(label)>_localctx.<label> = _prevctx;<endif>
try pushNewRecursionContext(_localctx, _startState, <parser.name>.RULE_<ruleName>)
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
_localctx = <currentAltLabel; format="cap">Context(  <ruleName; format="cap">Context(_parentctx,
_parentState))
<if(label)>
<if(isListLabel)>
(_localctx as! <currentAltLabel; format="cap">Context).<label>.append(_prevctx)
<else>
(_localctx as! <currentAltLabel; format="cap">Context).<label> = _prevctx
<endif>
<endif>
try pushNewRecursionContext(_localctx, _startState, <parser.name>.RULE_<ruleName>)
>>

recRuleReplaceContext(ctxName) ::= <<
_localctx = <ctxName>Context(_localctx)
_ctx = _localctx
_prevctx = _localctx
>>

recRuleSetPrevCtx() ::= <<
if _parseListeners != nil {
  try triggerExitRuleEvent()
}
_prevctx = _localctx
>>


LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
<if(lexerFile.genPackage)>
<!package <lexerFile.genPackage>;!>
<endif>
<namedActions.header>
import Antlr4

<lexer>
>>
```

```
Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
<accessLevelOpenOK(lexer)> class <lexer.name>: <superClass; null="Lexer"> {

 internal static var _decisionToDFA: [DFA] = {
        var decisionToDFA = [DFA]()
        let length = <lexer.name>._ATN.getNumberOfDecisions()
        for i in 0..\<length {
          <! decisionToDFA[i] = DFA(<lexer.name>._ATN.getDecisionState(i)!, i);!>
            decisionToDFA.append(DFA(<lexer.name>._ATN.getDecisionState(i)!, i))
        }
         return decisionToDFA
    }()

 internal static let _sharedContextCache = PredictionContextCache()

 <accessLevelNotOpen(lexer)>
 <if(lexer.tokens)>
 static let <lexer.tokens:{k | <k>=<lexer.tokens.(k)>}; separator=", ", wrap, anchor>
 <endif>

 <if(lexer.channels)>
 <accessLevelNotOpen(lexer)>
 static let <lexer.channels:{k | <k>=<lexer.channels.(k)>}; separator=", ", wrap, anchor>
 <endif>
 <if(rest(lexer.modes))>
 <accessLevelNotOpen(lexer)>
 static let <rest(lexer.modes):{m| <m>=<i>}; separator=", ", wrap, anchor>
 <endif>
 <accessLevelNotOpen(lexer)>
 static let channelNames: [String] = [
  "DEFAULT_TOKEN_CHANNEL", "HIDDEN"<if (lexer.channels)>, <lexer.channels:{c| "<c>"}; separator=", ",
wrap, anchor><endif>
 ]

 <accessLevelNotOpen(lexer)>
 static let modeNames: [String] = [
  <lexer.modes:{m| "<m>"}; separator=", ", wrap, anchor>
 ]

 <accessLevelNotOpen(lexer)>
 static let ruleNames: [String] = [
  <lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
 ]

 <vocabulary(lexer.literalNames, lexer.symbolicNames,
                accessLevelNotOpen(lexer))>

 <namedActions.members>
```

```
 override <accessLevelOpenOK(lexer)>
 func getVocabulary() -> Vocabulary {
  return <lexer.name>.VOCABULARY
 }

 <accessLevelNotOpen(lexer)>
 required init(_ input: CharStream) {
    RuntimeMetaData.checkVersion("<lexerFile.ANTLRVersion>", RuntimeMetaData.VERSION)
  super.init(input)
  _interp = LexerATNSimulator(self, <lexer.name>._ATN, <lexer.name>._decisionToDFA,
<lexer.name>._sharedContextCache)
 }

 override <accessLevelOpenOK(lexer)>
 func getGrammarFileName() -> String { return "<lexer.grammarFileName>" }

 override <accessLevelOpenOK(lexer)>
 func getRuleNames() -> [String] { return <lexer.name>.ruleNames }

 override <accessLevelOpenOK(lexer)>
 func getSerializedATN() -> String { return <lexer.name>._serializedATN }

 override <accessLevelOpenOK(lexer)>
 func getChannelNames() -> [String] { return <lexer.name>.channelNames }

 override <accessLevelOpenOK(lexer)>
 func getModeNames() -> [String] { return <lexer.name>.modeNames }

 override <accessLevelOpenOK(lexer)>
 func getATN() -> ATN { return <lexer.name>._ATN }

 <dumpActions(lexer, "", actionFuncs, sempredFuncs)>
 <atn>

 <accessLevelNotOpen(lexer)>
 static let _serializedATN: String = <lexer.name>ATN().jsonString

 <accessLevelNotOpen(lexer)>
 static let _ATN: ATN = ATNDeserializer().deserializeFromJson(_serializedATN)
}
>>

/** Don't need to define anything. The tool generates a XParserATN.swift file (and same for lexer)
*  which is referenced from static field _serializedATN.  This json string is passed to
*  deserializeFromJson(). Note this is not the "serialization as array of ints" that other targets
*  do. It is more or less the output of ATNPrinter which gets read back in.
*/
```

```
SerializedATN(model) ::= <<
>>

/** Using a type to init value map, try to init a type; if not in table
* must be an object, default value is "null".
*/
initValue(typeName) ::= <<
<SwiftTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".swift"
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/codegen/Swift/Swift.stg
No license file was found, but licenses were detected in source scan.

/*
[The "BSD licence"]
Copyright (c) 2006 Kay Roepke
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
```

```
/*
This file contains the actual layout of the messages emitted by ANTLR.
The text itself is coming out of the languages/*stg files, according to the chosen locale.
This file contains the default format ANTLR uses.
```
```

*/

location(file, line, column) ::= "<file>(<line>,<column>)"

message(id, text) ::= "error <id> : <text>"

report(location, message, type) ::= "<location> : <type> <message.id> : <message.text>"

wantsSingleLineMessage() ::= "true"

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/messages/formats/vs2005.stg
No license file was found, but licenses were detected in source scan.

/*
[The "BSD licence"]
Copyright (c) 2005-20012 Terence Parr
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ANTLRParser.java
No license file was found, but licenses were detected in source scan.

/*

```
* [The "BSD license"]
*  Copyright (c) 2016, Mike Lischke
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/tool/templates/codegen/Cpp/Files.stg
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.
* Use of this file is governed by the BSD 3-clause license that
* can be found in the LICENSE.txt file in the project root.
*/
/**
 [The "BSD license"]
 Copyright (c) 2011 Cay Horstmann
 All rights reserved.

 Redistribution and use in source and binary forms, with or without
 modification, are permitted provided that the following conditions
 are met:

 1. Redistributions of source code must retain the above copyright
```

notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/GraphicsSupport.java
No license file was found, but licenses were detected in source scan.

```
/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
```

```
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

phpTypeInitMap ::= [
 "int":"0",
 "long":"0",
 "float":"0.0",
 "double":"0.0",
 "boolean":"false",
 default:"null"
]

// args must be <object-model-object>, <fields-resulting-in-STs>

ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<parser>
>>

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage>;
<endif>
<header>
use Antlr\\Antlr4\\Runtime\\Tree\\ParseTreeListener;

/**
* This interface defines a complete listener for a parse tree produced by
* {@see <file.parserName>}.
*/
interface <file.grammarName>Listener extends ParseTreeListener {
 <file.listenerNames:{lname |
/**
<if(file.listenerLabelRuleNames.(lname))>
* Enter a parse tree produced by the `<lname>`
* labeled alternative in {@see <file.parserName>::<file.listenerLabelRuleNames.(lname)>()\}.
<else>
* Enter a parse tree produced by {@see <file.parserName>::<lname>()\}.
<endif>
* @param $context The parse tree.
*/
public function enter<lname; format="cap">(Context\\<lname; format="cap">Context $context) : void;
/**
<if(file.listenerLabelRuleNames.(lname))>
* Exit a parse tree produced by the `<lname>` labeled alternative
```

```
* in {@see <file.parserName>::<file.listenerLabelRuleNames.(lname)>()\}.
<else>
* Exit a parse tree produced by {@see <file.parserName>::<lname>()\}.
<endif>
* @param $context The parse tree.
*/
public function exit<lname; format="cap">(Context\\<lname; format="cap">Context $context) : void;};
separator="\n">
}
>>

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage>;
<endif>
<header>

use Antlr\\Antlr4\\Runtime\\ParserRuleContext;
use Antlr\\Antlr4\\Runtime\\Tree\\ErrorNode;
use Antlr\\Antlr4\\Runtime\\Tree\\TerminalNode;

/**
* This class provides an empty implementation of {@see <file.grammarName>Listener},
* which can be extended to create a listener which only needs to handle a subset
* of the available methods.
*/
class <file.grammarName>BaseListener implements <file.grammarName>Listener
{
 <file.listenerNames:{lname |
/**
* {@inheritdoc\}
*
* The default implementation does nothing.
*/
public function enter<lname; format="cap">(Context\\<lname; format="cap">Context $context) : void {\}

/**
* {@inheritdoc\}
*
* The default implementation does nothing.
*/
public function exit<lname; format="cap">(Context\\<lname; format="cap">Context $context) : void {\}};
separator="\n">

 /**
  * {@inheritdoc\}
  *
```

```
 * The default implementation does nothing.
 */
 public function enterEveryRule(ParserRuleContext $context) : void {}

 /**
  * {@inheritdoc\}
  *
  * The default implementation does nothing.
  */
 public function exitEveryRule(ParserRuleContext $context) : void {}

 /**
  * {@inheritdoc\}
  *
  * The default implementation does nothing.
  */
 public function visitTerminal(TerminalNode $node) : void {}

 /**
  * {@inheritdoc\}
  *
  * The default implementation does nothing.
  */
 public function visitErrorNode(ErrorNode $node) : void {}
}
>>

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage>;
<endif>

<header>
use Antlr\\Antlr4\\Runtime\\Tree\\ParseTreeVisitor;

/**
* This interface defines a complete generic visitor for a parse tree produced by {@see <file.parserName>}.
*/
interface <file.grammarName>Visitor extends ParseTreeVisitor
{
 <file.visitorNames:{lname |
/**
<if(file.visitorLabelRuleNames.(lname))>
* Visit a parse tree produced by the `<lname>` labeled alternative
* in {@see <file.parserName>::<file.visitorLabelRuleNames.(lname)>()\}.
<else>
* Visit a parse tree produced by {@see <file.parserName>::<lname>()\}.
```

```
<endif>
*
* @param Context\\<lname; format="cap">Context $context The parse tree.
*
* @return mixed The visitor result.
*/
public function visit<lname; format="cap">(Context\\<lname; format="cap">Context $context);};
separator="\n\n">
}
>>

BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage>;
<endif>
<header>
use Antlr\\Antlr4\\Runtime\\Tree\\AbstractParseTreeVisitor;

/**
* This class provides an empty implementation of {@see <file.grammarName>Visitor},
* which can be extended to create a visitor which only needs to handle a subset
* of the available methods.
*/
class <file.grammarName>BaseVisitor extends AbstractParseTreeVisitor implements <file.grammarName>Visitor
{
 <file.visitorNames:{lname |
/**
* {@inheritdoc\}
*
* The default implementation returns the result of calling
* {@see self::visitChildren()\} on `context`.
*/
public function visit<lname; format="cap">(Context\\<lname; format="cap">Context $context)
{
  return $this->visitChildren($context);
\}}; separator="\n\n">
}
>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
\<?php

/*
* Generated from <grammarFileName> by ANTLR <ANTLRVersion>
*/

>>
```

```
Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
namespace<if(file.genPackage)> <file.genPackage><endif> {
<if(namedActions.header)><namedActions.header><endif>
 use Antlr\\Antlr4\\Runtime\\Atn\\ATN;
 use Antlr\\Antlr4\\Runtime\\Atn\\ATNDeserializer;
 use Antlr\\Antlr4\\Runtime\\Atn\\ParserATNSimulator;
 use Antlr\\Antlr4\\Runtime\\Dfa\\DFA;
 use Antlr\\Antlr4\\Runtime\\Error\\Exceptions\\FailedPredicateException;
 use Antlr\\Antlr4\\Runtime\\Error\\Exceptions\\NoViableAltException;
 use Antlr\\Antlr4\\Runtime\\PredictionContexts\\PredictionContextCache;
 use Antlr\\Antlr4\\Runtime\\Error\\Exceptions\\RecognitionException;
 use Antlr\\Antlr4\\Runtime\\RuleContext;
 use Antlr\\Antlr4\\Runtime\\Token;
 use Antlr\\Antlr4\\Runtime\\TokenStream;
 use Antlr\\Antlr4\\Runtime\\Vocabulary;
 use Antlr\\Antlr4\\Runtime\\VocabularyImpl;
 use Antlr\\Antlr4\\Runtime\\RuntimeMetaData;
 use Antlr\\Antlr4\\Runtime\\Parser;
<if(namedActions.definitions)><namedActions.definitions><endif>

 final class <parser.name> extends <superClass; null="Parser">
 {
  <if(parser.tokens)>
  public const <parser.tokens:{k | <k> = <parser.tokens.(k)>}; separator=", ", wrap, anchor>;
  <endif>

  <if(parser.rules)>
  public const <parser.rules:{r | RULE_<r.name> = <r.index>}; separator=", ", wrap, anchor>;
  <endif>

  /**
   * @var array\<string>
   */
  public const RULE_NAMES = [
   <parser.ruleNames:{r | '<r>'}; separator=", ", wrap, anchor>
  ];

  <vocabulary(parser.literalNames, parser.symbolicNames)>

  <atn>
  protected static $atn;
  protected static $decisionToDFA;
  protected static $sharedContextCache;
  <if(namedActions.members)>
```

```
<namedActions.members>
<endif>


<parser:(ctor)()>

private static function initialize() : void
{
 if (self::$atn !== null) {
  return;
 }

 RuntimeMetaData::checkVersion('<file.ANTLRVersion>', RuntimeMetaData::VERSION);

 $atn = (new ATNDeserializer())->deserialize(self::SERIALIZED_ATN);

 $decisionToDFA = [];
 for ($i = 0, $count = $atn->getNumberOfDecisions(); $i \< $count; $i++) {
  $decisionToDFA[] = new DFA($atn->getDecisionState($i), $i);
 }

 self::$atn = $atn;
 self::$decisionToDFA = $decisionToDFA;
 self::$sharedContextCache = new PredictionContextCache();
}

public function getGrammarFileName() : string
{
 return "<parser.grammarFileName>";
}

public function getRuleNames() : array
{
 return self::RULE_NAMES;
}

public function getSerializedATN() : string
{
 return self::SERIALIZED_ATN;
}

public function getATN() : ATN
{
 return self::$atn;
}

public function getVocabulary() : Vocabulary
```

```
    {
        static $vocabulary;

  return $vocabulary = $vocabulary ?? new VocabularyImpl(self::LITERAL_NAMES, self::SYMBOLIC_NAMES);
    }
<if(funcs)>

 <funcs; separator="\n\n">
<endif>
<if(sempredFuncs)>

 public function sempred(?RuleContext $localContext, int $ruleIndex, int $predicateIndex) : bool
 {
  switch ($ruleIndex) {
  <parser.sempredFuncs.values:{f|
 case <f.ruleIndex>:
  return $this->sempred<f.name; format="cap">($localContext, $predicateIndex);}; separator="\n\n">

   default:
    return true;
  }
 }

 <sempredFuncs.values; separator="\n\n">
<endif>
 }
}

namespace <if(file.genPackage)><file.genPackage>\\<endif>Context {
 use Antlr\\Antlr4\\Runtime\\ParserRuleContext;
 use Antlr\\Antlr4\\Runtime\\Token;
 use Antlr\\Antlr4\\Runtime\\Tree\\ParseTreeVisitor;
 use Antlr\\Antlr4\\Runtime\\Tree\\TerminalNode;
 use Antlr\\Antlr4\\Runtime\\Tree\\ParseTreeListener;
 use <if(file.genPackage)><file.genPackage>\\<endif><parser.name>;
 <if (file.genVisitor)>use <if(file.genPackage)><file.genPackage>\\<endif><file.grammarName>Visitor;<endif>
 <if (file.genListener)>use <if(file.genPackage)><file.genPackage>\\<endif><file.grammarName>Listener;<endif>
 <namedActions.contexts>

 <funcs :{ func | <func.ruleCtx><if(func.altLabelCtxs)>

<func.altLabelCtxs:{l | <func.altLabelCtxs.(l)>}; separator="\n\n"><endif> }; separator="\n\n">
}
>>

vocabulary(literalNames, symbolicNames) ::= <<
/**
* @var array\<string|null>
```

```
*/
private const LITERAL_NAMES = [
   <literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
];

/**
* @var array\<string>
*/
private const SYMBOLIC_NAMES = [
   <symbolicNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
];
>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>

public function action(?RuleContext $localContext, int $ruleIndex, int $actionIndex) : void
{
 switch ($ruleIndex) {
 <recog.actionFuncs.values:{f|
 case <f.ruleIndex>:
  $this->action<f.name; format="cap">($localContext, $actionIndex);
  break;}; separator="\n\n">
 }
}

<actionFuncs.values; separator="\n">
<endif>
<if(sempredFuncs)>

public function sempred(?RuleContext $localContext, int $ruleIndex, int $predicateIndex) : bool
{
 switch ($ruleIndex) {
 <recog.sempredFuncs.values:{f|
   case <f.ruleIndex>:
       return $this->sempred<f.name; format="cap">($localContext, $predicateIndex);}; separator="\n\n">
 }

 return true;
}
<sempredFuncs.values; separator="\n\n">
<endif>
>>

parser_ctor(p) ::= <<
public function __construct(TokenStream $input)
{
 parent::__construct($input);
```

```
 self::initialize();

 $this->interp = new ParserATNSimulator($this, self::$atn, self::$decisionToDFA, self::$sharedContextCache);
}
>>

/**
* This generates a private method since the actionIndex is generated, making
* an overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<
private function action<r.name; format="cap">(?<r.ctxType> $localContext, int $actionIndex) : void
{
 switch ($actionIndex) {
 <actions:{index|
 case <index>:
 <actions.(index)>

 break;}; separator="\n\n">
 }
}
>>

/**
* This generates a private method since the predicateIndex is generated, making
* an overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
private function sempred<r.name; format="cap">(?Context\\<r.ctxType> $localContext, int $predicateIndex) : bool
{
 switch ($predicateIndex) {
 <actions:{index|
   case <index>:
       return <actions.(index)>;}; separator="\n\n">
 }

 return true;
}
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,exceptions,postamble)
::= <<
/**
* @throws RecognitionException
*/
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><endif>public function <currentRule.name>(<args;
separator=","> : Context\\<currentRule.ctxType>
```

```
{
  $localContext = new Context\\<currentRule.ctxType>($this->ctx, $this->getState()<currentRule.args:{a | ,
$<a.name>}>);

  $this->enterRule($localContext, <currentRule.startState>, self::RULE_<currentRule.name>);
  <namedActions.init>
  <locals; separator="\n">

  try {
     <code>
     <postamble; separator="\n">
     <namedActions.after>
  }<if(exceptions)><exceptions; separator="\n"><else> catch (RecognitionException $exception) {
     $localContext->exception = $exception;
     $this->errorHandler->reportError($this, $exception);
     $this->errorHandler->recover($this, $exception);
  }<endif> finally {
     <finallyAction>
     $this->exitRule();
  }

  return $localContext;
}
>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamb
le) ::= <<
/**
* @throws RecognitionException
*/
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><endif>public function <currentRule.name>(<args;
separator=", ">) : Context\\<currentRule.ctxType>
{
 return $this->recursive<currentRule.name; format="cap">(0<currentRule.args:{a | , <a.name>}>);
}

/**
* @throws RecognitionException
*/
private function recursive<currentRule.name; format="cap">(int $precedence<args:{a | , <a>}>) :
Context\\<currentRule.ctxType>
{
 $parentContext = $this->ctx;
 $parentState = $this->getState();
 $localContext = new Context\\<currentRule.ctxType>($this->ctx, $parentState<currentRule.args:{a | ,
<a.name>}>);
 $previousContext = $localContext;
 $startState = <currentRule.startState>;
```

```
 $this->enterRecursionRule($localContext, <currentRule.startState>, self::RULE_<currentRule.name>,
$precedence);
 <namedActions.init>
 <locals; separator="\n">

 try {
  <code>
  <postamble; separator="\n">
  <namedActions.after>
 } catch (RecognitionException $exception) {
  $localContext->exception = $exception;
  $this->errorHandler->reportError($this, $exception);
  $this->errorHandler->recover($this, $exception);
 } finally {
  <finallyAction>
  $this->unrollRecursionContexts($parentContext);
 }

 return $localContext;
}
>>

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>$localContext = new
Context\\<currentOuterMostAltCodeBlock.altLabel; format="cap">Context($localContext);<endif>
$this->enterOuterAlt($localContext, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>

LL1AltBlock(choice, preamble, alts, error) ::= <<
$this->setState(<choice.stateNumber>);
$this->errorHandler->sync($this);
<if(choice.label)><labelref(choice.label)> = $this->input->LT(1);<endif>
<preamble; separator="\n">

switch ($this->input->LA(1)) {
   <choice.altLook,alts:{look,alt| <cases(ttypes=look)>
 <alt>
 break;}; separator="\n\n">

default:
 <error>
```

```
}
>>

LL1OptionalBlock(choice, alts, error) ::= <<
$this->setState(<choice.stateNumber>);
$this->errorHandler->sync($this);

switch ($this->input->LA(1)) {
  <choice.altLook,alts:{look,alt| <cases(ttypes=look)>
 <alt>
 break;}; separator="\n\n">

default:
 break;
}
>>

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
$this->setState(<choice.stateNumber>);
$this->errorHandler->sync($this);
<preamble; separator="\n">

if (<expr>) {
 <alts; separator="\n">
}
>>

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
$this->setState(<choice.stateNumber>);
$this->errorHandler->sync($this);

<preamble; separator="\n">
while (<loopExpr>) {
 <alts; separator="\n">
 $this->setState(<choice.loopBackStateNumber>);
 $this->errorHandler->sync($this);
 <iteration>
}
>>

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
$this->setState(<choice.blockStartStateNumber>); <! alt block decision !>
$this->errorHandler->sync($this);

<preamble; separator="\n">
do {
 <alts; separator="\n">
 $this->setState(<choice.stateNumber>); <! loopback/exit decision !>
```

```
 $this->errorHandler->sync($this);
 <iteration>
} while (<loopExpr>);
>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
$this->setState(<choice.stateNumber>);
$this->errorHandler->sync($this);
<if(choice.label)><labelref(choice.label)> = $this->input->LT(1);<endif>
<preamble; separator="\n">

switch ($this->getInterpreter()->adaptivePredict($this->input, <choice.decision>, $this->ctx)) {
 <alts:{alt |
case <i>:
   <alt>
break;}; separator="\n\n">
}
>>

OptionalBlock(choice, alts, error) ::= <<
$this->setState(<choice.stateNumber>);
$this->errorHandler->sync($this);

switch ($this->getInterpreter()->adaptivePredict($this->input, <choice.decision>, $this->ctx)) {
<alts:{alt |
   case <i><if(!choice.ast.greedy)>+1<endif>:
     <alt>
 break;}; separator="\n\n">
}
>>

StarBlock(choice, alts, sync, iteration) ::= <<
$this->setState(<choice.stateNumber>);
$this->errorHandler->sync($this);

$alt = $this->getInterpreter()->adaptivePredict($this->input, <choice.decision>, $this->ctx);

while ($alt !== <choice.exitAlt> && $alt !== ATN::INVALID_ALT_NUMBER) {
 if ($alt === 1<if(!choice.ast.greedy)>+1<endif>) {
  <iteration>
  <alts> <! should only be one !>
 }

 $this->setState(<choice.loopBackStateNumber>);
 $this->errorHandler->sync($this);
```

```
 $alt = $this->getInterpreter()->adaptivePredict($this->input, <choice.decision>, $this->ctx);
}
>>

PlusBlock(choice, alts, error) ::= <<
$this->setState(<choice.blockStartStateNumber>); <! alt block decision !>
$this->errorHandler->sync($this);

$alt = 1<if(!choice.ast.greedy)>+1<endif>;

do {
 switch ($alt) {
 <alts:{alt|
case <i><if(!choice.ast.greedy)>+1<endif>:
 <alt>
 break;}; separator="\n\n">
 default:
  <error>
 }

 $this->setState(<choice.loopBackStateNumber>); <! loopback/exit decision !>
 $this->errorHandler->sync($this);

 $alt = $this->getInterpreter()->adaptivePredict($this->input, <choice.decision>, $this->ctx);
} while ($alt !== <choice.exitAlt> && $alt !== ATN::INVALID_ALT_NUMBER);
>>

Sync(s) ::= "sync(<s.expecting.name>);"

ThrowNoViableAlt(t) ::= "throw new NoViableAltException($this);"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) === 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShiftVar(s.varName, bits.shift)>})> && ((1 \<\< <offsetShiftVar(s.varName,
bits.shift)>) & (<bits.ttypes:{ttype | (1 \<\< <offsetShiftConst(ttype, bits.shift)>)}; separator=" | ">)) !== 0)
%>

isZero ::= [
```

```
"0":true,
default:false
]

offsetShiftVar(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>($<shiftAmount> - <offset>)<else>$<shiftAmount><endif>
%>
offsetShiftConst(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(self::<shiftAmount> - <offset>)<else>self::<shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | $<s.varName> === self::<ttype>}; separator=" || ">
%>

cases(ttypes) ::= <<
<ttypes:{t | case self::<t>:}; separator="\n">
>>

InvokeRule(r, argExprsChunks) ::= <<
$this->setState(<r.stateNumber>);
<if(r.labels)><r.labels:{l | <labelref(l)> = }><endif>$this-><if(r.ast.options.p)>recursive<r.name;
format="cap"><else><r.name><endif>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif>
<argExprsChunks>);
>>

MatchToken(m) ::= <<
$this->setState(<m.stateNumber>);
<if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>$this->match(self::<m.name>);
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
$this->setState(<m.stateNumber>);

<if(m.labels)><m.labels:{l | <labelref(l)> = }>$this->input->LT(1);<endif>
<capture>

if (<if(invert)>$<m.varName> \<= 0 || <else>!<endif>(<expr>)) {
    <if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>$this->errorHandler->recoverInline($this);
} else {
 if ($this->input->LA(1) === Token::EOF) {
    $this->matchedEOF = true;
  }
```

```
 $this->errorHandler->reportMatch($this);
 $this->consume();
}
>>

Wildcard(w) ::= <<
$this->setState(<w.stateNumber>);
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>$this->matchWildcard();
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
$this->setState(<p.stateNumber>);

if (!(<chunks>)) {
  throw new FailedPredicateException($this, <p.predicate><if(failChunks)>, <failChunks><elseif(p.msg)>,
<p.msg><endif>);
}
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
 <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand()  ::= "$this->skip();"
LexerMoreCommand()  ::= "$this->more();"
LexerPopModeCommand() ::= "$this->popMode();"

LexerTypeCommand(arg, grammar)      ::= "$this->type = <arg>;"
LexerChannelCommand(arg, grammar)   ::= "$this->channel = <arg>;"
LexerModeCommand(arg, grammar)      ::= "$this->mode = <arg>;"
LexerPushModeCommand(arg, grammar)  ::= "$this->pushMode(<arg>);"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "$localContext-><a.name>"
LocalRef(a) ::= "$localContext-><a.name>"
RetValueRef(a) ::= "$localContext-><a.name>"
```

```
QRetValueRef(a) ::= "<ctx(a)>-><a.dict>-><a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>-><t.name>"
LabelRef(t) ::= "<ctx(t)>-><t.name>"
ListLabelRef(t) ::= "<ctx(t)>-><ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>-><s.name> = <rhsChunks>;"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(<ctx(t)>-><t.label> !== null ? <ctx(t)>-><t.label>->getText() : null)"
TokenPropertyRef_type(t) ::= "(<ctx(t)>-><t.label> !== null ? <ctx(t)>-><t.label>->getType() : 0)"
TokenPropertyRef_line(t) ::= "(<ctx(t)>-><t.label> !== null ? <ctx(t)>-><t.label>->getLine() : 0)"
TokenPropertyRef_pos(t) ::= "(<ctx(t)>-><t.label> !== null ? <ctx(t)>-><t.label>->getCharPositionInLine() : 0)"
TokenPropertyRef_channel(t) ::= "(<ctx(t)>-><t.label> !== null ? <ctx(t)>-><t.label>->getChannel() : 0)"
TokenPropertyRef_index(t) ::= "(<ctx(t)>-><t.label> !== null ? <ctx(t)>-><t.label>->getTokenIndex() : 0)"
TokenPropertyRef_int(t) ::= "(<ctx(t)>-><t.label> !== null ? (int) <ctx(t)>-><t.label>->getText() : 0)"

RulePropertyRef_start(r) ::= "(<ctx(r)>-><r.label> !== null ? (<ctx(r)>-><r.label>->start) : null)"
RulePropertyRef_stop(r)  ::= "(<ctx(r)>-><r.label> !== null ? (<ctx(r)>-><r.label>->stop) : null)"
RulePropertyRef_text(r)  ::= "(<ctx(r)>-><r.label> !== null ? $this->input->getTextByTokens(<ctx(r)>-><r.label>-
>start, <ctx(r)>-><r.label>->stop) : null)"
RulePropertyRef_ctx(r)  ::= "<ctx(r)>-><r.label>"
RulePropertyRef_parser(r):= "\$this"

ThisRulePropertyRef_start(r) ::= "$localContext->start"
ThisRulePropertyRef_stop(r)  ::= "$localContext->stop"
ThisRulePropertyRef_text(r)  ::= "$this->input->getTextByTokens($localContext->start, $this->input->LT(-1))"
ThisRulePropertyRef_ctx(r)  ::= "$localContext"
ThisRulePropertyRef_parser(r):= "$this"

NonLocalAttrRef(s)        ::= "\$this->getInvokingContext(<s.ruleIndex>)-><s.name>"
SetNonLocalAttr(s, rhsChunks) ::= "\$this->getInvokingContext(<s.ruleIndex>)-><s.name> = <rhsChunks>;"

AddToLabelList(a) ::= "<ctx(a.label)>-><a.listName>[] = <labelref(a.label)>;"

TokenDecl(t) ::= "<TokenLabelType()> $<t.name>"
TokenTypeDecl(t) ::= ""
TokenListDecl(t) ::= "array $<t.name> = []"
RuleContextDecl(r) ::= "<r.ctxName> $<r.name>"
RuleContextListDecl(rdecl) ::= "array $<rdecl.name> = []"
AttributeDecl(d) ::= "<d.type> $<d.name><if(d.initValue)> = <d.initValue><endif>"

PropertiesDecl(struct) ::= <<
<if(struct.tokenListDecls)>
 <struct.tokenListDecls : {d | /**
* @var array\<Token>|null $<d.name>
*/
```

```
public $<d.name>;}; separator="\n\n">
<endif>
<if(struct.tokenDecls)>
<if(struct.tokenListDecls)>

<endif>
 <struct.tokenDecls : {d | /**
* @var <TokenLabelType()>|null $<d.name>
*/
public $<d.name>;}; separator="\n\n">
<endif>
<if(struct.ruleContextDecls)>
<if(struct.tokenListDecls || struct.tokenDecls)>

<endif>
 <struct.ruleContextDecls : {d | /**
* @var <d.ctxName>|null $<d.name>
*/
public $<d.name>;}; separator="\n\n">
<endif>
<if(struct.ruleContextListDecls)>
<if(struct.tokenListDecls || struct.tokenDecls || struct.ruleContextDecls)>

<endif>
 <struct.ruleContextListDecls : {d | /**
* @var array\<<d.ctxName>\>|null $<d.name>
*/
public $<d.name>;}; separator="\n\n">
<endif>
<if(struct.attributeDecls)>
<if(struct.tokenListDecls || struct.tokenDecls || struct.ruleContextDecls || struct.ruleContextListDecls)>

<endif>
 <struct.attributeDecls : {d | /**
* @var <d.type><if(!d.initValue)>|null<endif> $<d.name>
*/
public $<d.name><if(d.initValue)> = <d.initValue><endif>;}; separator="\n\n">
<endif>

>>

ContextTokenGetterDecl(t) ::= <<
public function <t.name>() : ?TerminalNode
{
  return $this->getToken(<parser.name>::<t.name>, 0);
}
>>
```

```
ContextTokenListGetterDecl(t)  ::= <<
>>

ContextTokenListIndexedGetterDecl(t)  ::= <<
/**
* @return array\<TerminalNode>|TerminalNode|null
*/
public function <t.name>(?int $index = null)
{
 if ($index === null) {
  return $this->getTokens(<parser.name>::<t.name>);
 }

   return $this->getToken(<parser.name>::<t.name>, $index);
}
>>

ContextRuleGetterDecl(r)       ::= <<
public function <r.name>() : ?<r.ctxName>
{
 return $this->getTypedRuleContext(<r.ctxName>::class, 0);
}
>>

ContextRuleListGetterDecl(r)   ::= <<
>>

ContextRuleListIndexedGetterDecl(r)   ::= <<
/**
* @return array\<<r.ctxName>\>|<r.ctxName>|null
*/
public function <r.name>(?int $index = null)
{
 if ($index === null) {
  return $this->getTypedRuleContexts(<r.ctxName>::class);
 }

   return $this->getTypedRuleContext(<r.ctxName>::class, $index);
}
>>

LexerRuleContext() ::= "RuleContext"

/**
* The rule context name is the rule followed by a suffix; e.g., r becomes rContext.
*/
RuleContextNameSuffix() ::= "Context"
```

```
ImplicitTokenLabel(tokenName) ::= "<tokenName>"
ImplicitRuleLabel(ruleName)   ::= "<ruleName>"
ImplicitSetLabel(id)    ::= "_tset<id>"
ListLabelName(label)    ::= "<label>"

CaptureNextToken(d) ::= "$<d.varName> = \$this->input->LT(1);"
CaptureNextTokenType(d) ::= "$<d.varName> = $this->input->LA(1);"

StructDecl(struct,ctorAttrs,attrs,getters,dispatchMethods,interfaces,extensionMembers) ::= <<
class <struct.name> extends
<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)> implements
<interfaces; separator=", "><endif>
{
<PropertiesDecl(struct)>
 public function __construct(?ParserRuleContext $parent, ?int $invokingState = null<ctorAttrs:{a | , ?<a> = null}>)
 {
  parent::__construct($parent, $invokingState);
<if(struct.ctorAttrs)>

  <struct.ctorAttrs:{a | $this-><a.name> = $<a.name> ?? $this-><a.name>;}; separator="\n">
<endif>
 }

 public function getRuleIndex() : int
 {
    return <parser.name>::RULE_<struct.derivedFromName>;
  }
<if(getters)>

  <getters:{g | <g>}; separator="\n\n">
<endif>
<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
 public function copyFrom(ParserRuleContext $context) : void
 {
  parent::copyFrom($context);

  <struct.attrs:{a | $this-><a.name> = $context-><a.name>;}; separator="\n">
 }
<endif>
<if(dispatchMethods)>

 <dispatchMethods; separator="\n\n">
<endif>
<if(extensionMembers)>

 <extensionMembers; separator="\n\n">
<endif>
}
```

```
>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
class <struct.name> extends <struct.parentRule; format="cap">Context
{
<PropertiesDecl(struct)>
 public function __construct(<struct.parentRule; format="cap">Context $context)
 {
    parent::__construct($context);

    $this->copyFrom($context);
  }
<if(getters)>

  <getters:{g | <g>}; separator="\n\n">
<endif>
<if(dispatchMethods)>

 <dispatchMethods; separator="\n\n">
<endif>
}
>>

ListenerDispatchMethod(method) ::= <<
public function <if(method.isEnter)>enter<else>exit<endif>Rule(ParseTreeListener $listener) : void
{
 if ($listener instanceof <parser.grammarName>Listener) {
    $listener-><if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">($this);
  }
}
>>

VisitorDispatchMethod(method) ::= <<
public function accept(ParseTreeVisitor $visitor)
{
 if ($visitor instanceof <parser.grammarName>Visitor) {
    return $visitor->visit<struct.derivedFromName; format="cap">($this);
  }

 return $visitor->visitChildren($this);
}
>>

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>$localContext-><endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "$localContext"
```

```
// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec)  ::= "\$this->precpred(\$this->ctx, <opPrec>)"

recRuleSetReturnAction(src,name)   ::= "\$<name> = \$<src>-><name>;"

recRuleSetStopToken()              ::= "$this->ctx->stop = $this->input->LT(-1);"

recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<
$localContext = new Context\\<ctxName>Context($parentContext, $parentState);
<if(label)>
<if(isListLabel)>
$localContext-><label>[] = $previousContext;
<else>
$localContext-><label> = $previousContext;
<endif>
<endif>

$this->pushNewRecursionContext($localContext, $startState, self::RULE_<ruleName>);
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
$localContext = new Context\\<currentAltLabel; format="cap">Context(new Context\\<ruleName;
format="cap">Context($parentContext, $parentState));
<if(label)>
<if(isListLabel)>
$localContext-><label>[] = $previousContext;
<else>
$localContext-><label> = $previousContext;
<endif>
<endif>

$this->pushNewRecursionContext($localContext, $startState, self::RULE_<ruleName>);
>>

recRuleReplaceContext(ctxName) ::= <<
$localContext = new Context\\<ctxName>Context($localContext);
$this->ctx = $localContext;
$previousContext = $localContext;
>>

recRuleSetPrevCtx() ::= <<
if ($this->getParseListeners() !== null) {
  $this->triggerExitRuleEvent();
}

$previousContext = $localContext;
>>
```

```
LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
<lexer>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
namespace<if(lexerFile.genPackage)> <lexerFile.genPackage><endif> {
<if(namedActions.header)><namedActions.header><endif>
 use Antlr\\Antlr4\\Runtime\\Atn\\ATNDeserializer;
 use Antlr\\Antlr4\\Runtime\\Atn\\LexerATNSimulator;
 use Antlr\\Antlr4\\Runtime\\Lexer;
 use Antlr\\Antlr4\\Runtime\\CharStream;
 use Antlr\\Antlr4\\Runtime\\PredictionContexts\\PredictionContextCache;
 use Antlr\\Antlr4\\Runtime\\RuleContext;
 use Antlr\\Antlr4\\Runtime\\Atn\\ATN;
 use Antlr\\Antlr4\\Runtime\\Dfa\\DFA;
 use Antlr\\Antlr4\\Runtime\\Vocabulary;
 use Antlr\\Antlr4\\Runtime\\RuntimeMetaData;
 use Antlr\\Antlr4\\Runtime\\VocabularyImpl;
<if(namedActions.definitions)><namedActions.definitions><endif>

 final class <lexer.name> extends <superClass; null="Lexer">
 {
  <if(lexer.tokens)>
  public const <lexer.tokens:{k | <k> = <lexer.tokens.(k)>}; separator=", ", wrap, anchor>;
  <endif>

  <if(lexer.channels)>
  public const <lexer.channels:{c | <c> = <lexer.channels.(c)>}; separator=", ", wrap, anchor>;
  <endif>

  <if(rest(lexer.modes))>
  public const <rest(lexer.modes):{m | <m>=<i>}; separator=", ", wrap, anchor>;
  <endif>

  /**
   * @var array\<string>
   */
  public const CHANNEL_NAMES = [
   'DEFAULT_TOKEN_CHANNEL', 'HIDDEN'<if (lexer.channels)>, <lexer.channels:{c| '<c>'}; separator=", ",
wrap, anchor><endif>
  ];

  /**
   * @var array\<string>
   */
```

```
 public const MODE_NAMES = [
  <lexer.modes:{m| '<m>'}; separator=", ", wrap, anchor>
 ];

 /**
  * @var array\<string>
  */
 public const RULE_NAMES = [
  <lexer.ruleNames:{r | '<r>'}; separator=", ", wrap, anchor>
 ];

 <vocabulary(lexer.literalNames, lexer.symbolicNames)>

 <atn>
 protected static $atn;
 protected static $decisionToDFA;
 protected static $sharedContextCache;
<if(namedActions.members)>

 <namedActions.members>
<endif>

 public function __construct(CharStream $input)
 {
  parent::__construct($input);

  self::initialize();

  $this->interp = new LexerATNSimulator($this, self::$atn, self::$decisionToDFA, self::$sharedContextCache);
 }

 private static function initialize() : void
 {
  if (self::$atn !== null) {
   return;
  }

  RuntimeMetaData::checkVersion('<lexerFile.ANTLRVersion>', RuntimeMetaData::VERSION);

  $atn = (new ATNDeserializer())->deserialize(self::SERIALIZED_ATN);

  $decisionToDFA = [];
  for ($i = 0, $count = $atn->getNumberOfDecisions(); $i \< $count; $i++) {
   $decisionToDFA[] = new DFA($atn->getDecisionState($i), $i);
  }

  self::$atn = $atn;
  self::$decisionToDFA = $decisionToDFA;
```

```
 self::$sharedContextCache = new PredictionContextCache();
}

public static function vocabulary() : Vocabulary
{
 static $vocabulary;

 return $vocabulary = $vocabulary ?? new VocabularyImpl(self::LITERAL_NAMES, self::SYMBOLIC_NAMES);
}

public function getGrammarFileName() : string
{
 return '<lexer.grammarFileName>';
}

public function getRuleNames() : array
{
 return self::RULE_NAMES;
}

public function getSerializedATN() : string
{
 return self::SERIALIZED_ATN;
}

/**
 * @return array\<string>
 */
public function getChannelNames() : array
{
 return self::CHANNEL_NAMES;
}

/**
 * @return array\<string>
 */
public function getModeNames() : array
{
 return self::MODE_NAMES;
}

public function getATN() : ATN
{
 return self::$atn;
}

public function getVocabulary() : Vocabulary
{
```

```
  return self::vocabulary();
 }
 <dumpActions(lexer, "", actionFuncs, sempredFuncs)>
}
}
>>

SerializedATN(model) ::= <<
<if(rest(model.segments))>
/**
* @var string
*/
private const SERIALIZED_ATN =
 <model.segments:{segment| "<segment; wrap={" .<\n>"}>"}; separator=" .\n">;
<else>
/**
* @var string
*/
private const SERIALIZED_ATN =
 "<model.serialized; wrap={" .<\n>    "}>";
<endif>
>>

/**
* Using a type to init value map, try to init a type; if not in table
* must be an object, default value is `null`.
*/
initValue(typeName) ::= <<
<phpTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".php"

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/tool/templates/codegen/PHP/PHP.stg
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
```

```
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

tree grammar BlockSetTransformer;
options {
 language     = Java;
 tokenVocab   = ANTLRParser;
 ASTLabelType = GrammarAST;
 output   = AST;
 filter   = true;
}

@header {
package org.antlr.v4.parse;
import org.antlr.v4.misc.Utils;
import org.antlr.v4.misc.*;
import org.antlr.v4.tool.*;
import org.antlr.v4.tool.ast.*;
import java.util.List;
import java.util.Set;
import java.util.HashSet;
import java.util.ArrayList;
import org.antlr.v4.runtime.misc.IntervalSet;
}

@members {
public String currentRuleName;
public GrammarAST currentAlt;
public Grammar g;
public BlockSetTransformer(TreeNodeStream input, Grammar g) {
  this(input, new RecognizerSharedState());
```

```
  this.g = g;
}
}

topdown
  : ^(RULE (id=TOKEN_REF|id=RULE_REF) {currentRuleName=$id.text;} .+)
  | setAlt
  | ebnfBlockSet
  | blockSet
 ;

setAlt
 : {inContext("RULE BLOCK")}?
  ALT {currentAlt = $start;}
 ;

// (BLOCK (ALT (+ (BLOCK (ALT INT) (ALT ID)))))
ebnfBlockSet
@after {
 GrammarTransformPipeline.setGrammarPtr(g, $tree);
}
 : ^(ebnfSuffix blockSet) -> ^(ebnfSuffix ^(BLOCK<BlockAST> ^(ALT<AltAST> blockSet)))
 ;

ebnfSuffix
@after {$tree = (GrammarAST)adaptor.dupNode($start);}
 : OPTIONAL
  | CLOSURE
  | POSITIVE_CLOSURE
 ;

blockSet
@init {
boolean inLexer = Grammar.isTokenName(currentRuleName);
}
@after {
 GrammarTransformPipeline.setGrammarPtr(g, $tree);
}
 : {inContext("RULE")}? // top-level: rule block and > 1 alt
  ^(BLOCK ^(alt=ALT elementOptions? {((AltAST)$alt).altLabel==null}? setElement[inLexer]) ( ^(ALT
elementOptions? setElement[inLexer]) )+)
  -> ^(BLOCK<BlockAST>[$BLOCK.token] ^(ALT<AltAST>[$BLOCK.token,"ALT"] ^(SET[$BLOCK.token,
"SET"] setElement+)))
 | {!inContext("RULE")}? // if not rule block and > 1 alt
  ^(BLOCK ^(ALT elementOptions? setElement[inLexer]) ( ^(ALT elementOptions? setElement[inLexer]) )+)
  -> ^(SET[$BLOCK.token, "SET"] setElement+)
 ;
```

```
setElement[boolean inLexer]
@after {
 GrammarTransformPipeline.setGrammarPtr(g, $tree);
}
 : ( ^(a=STRING_LITERAL elementOptions) {!inLexer ||
CharSupport.getCharValueFromGrammarCharLiteral($a.getText())!=-1}?
  |   a=STRING_LITERAL {!inLexer || CharSupport.getCharValueFromGrammarCharLiteral($a.getText())!=-1}?
  | {!inLexer}?=> ^(TOKEN_REF elementOptions)
  | {!inLexer}?=>   TOKEN_REF
  | {inLexer}?=>  ^(RANGE a=STRING_LITERAL b=STRING_LITERAL)
   {CharSupport.getCharValueFromGrammarCharLiteral($a.getText())!=-1 &&
    CharSupport.getCharValueFromGrammarCharLiteral($b.getText())!=-1}?
  )
 ;

elementOptions
 : ^(ELEMENT_OPTIONS elementOption*)
 ;

elementOption
 : ID
 | ^(ASSIGN id=ID v=ID)
 | ^(ASSIGN ID v=STRING_LITERAL)
 | ^(ASSIGN ID v=ACTION)
 | ^(ASSIGN ID v=INT)
 ;

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/parse/BlockSetTransformer.g
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  Copyright (c) 2014 Eric Vergnaud
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
```

```
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

/** ANTLR tool checks output templates are compatible with tool code generation.
*  For now, a simple string match used on x.y of x.y.z scheme.
*  Must match Tool.VERSION during load to templates.
*
*  REQUIRED.
*/

pythonTypeInitMap ::= [
 "bool":"False",
 "int":"0",
 "float":"0.0",
 "str":"",
 default:"None" // anything other than a primitive type is an object
]

// args must be <object-model-object>, <fields-resulting-in-STs>

ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
# encoding: utf-8
from __future__ import print_function
from antlr4 import *
from io import StringIO
import sys

<namedActions.header>
<parser>

>>

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
from antlr4 import *
```

```
<header>

# This class defines a complete listener for a parse tree produced by <file.parserName>.
class <file.grammarName>Listener(ParseTreeListener):

  <file.listenerNames:{lname |
# Enter a parse tree produced by <file.parserName>#<lname>.
def enter<lname; format="cap">(self, ctx):
  pass

# Exit a parse tree produced by <file.parserName>#<lname>.
def exit<lname; format="cap">(self, ctx):
  pass

}; separator="\n">

>>


VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
from antlr4 import *
<header>

# This class defines a complete generic visitor for a parse tree produced by <file.parserName>.

class <file.grammarName>Visitor(ParseTreeVisitor):

  <file.visitorNames:{lname |
# Visit a parse tree produced by <file.parserName>#<lname>.
def visit<lname; format="cap">(self, ctx):
  return self.visitChildren(ctx)

}; separator="\n">

>>


fileHeader(grammarFileName, ANTLRVersion) ::= <<
# Generated from <grammarFileName> by ANTLR <ANTLRVersion>
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
<if(superClass)>
```

```
if __name__ is not None and "." in __name__:
  from .<superClass> import <superClass>
else:
  from <superClass> import <superClass>

<endif>
<atn>

class <parser.name> ( <if(superClass)><superClass><else>Parser<endif> ):

  grammarFileName = "<parser.grammarFileName>"

  atn = ATNDeserializer().deserialize(serializedATN())

  decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]

  sharedContextCache = PredictionContextCache()

  literalNames = [ <parser.literalNames:{t | u<t>}; null="u\"\<INVALID>\"", separator=", ", wrap, anchor> ]

  symbolicNames = [ <parser.symbolicNames:{t | u<t>}; null="u\"\<INVALID>\"", separator=", ", wrap, anchor> ]

  <if(parser.rules)>
  <parser.rules:{r | RULE_<r.name> = <r.index>}; separator="\n", wrap, anchor>
  <endif>

  ruleNames =  [ <parser.ruleNames:{r | u"<r>"}; separator=", ", wrap, anchor> ]

  EOF = <TokenLabelType()>.EOF
  <if(parser.tokens)>
  <parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator="\n", wrap, anchor>
  <endif>

  <parser:(ctor)()>

  <namedActions.members>

  <funcs; separator="\n">


<if(sempredFuncs)>
  def sempred(self, localctx, ruleIndex, predIndex):
      if self._predicates == None:
          self._predicates = dict()
<parser.sempredFuncs.values:{f |
      self._predicates[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n        ">
      pred = self._predicates.get(ruleIndex, None)
      if pred is None:
```

```
            raise Exception("No predicate with index:" + str(ruleIndex))
        else:
            return pred(localctx, predIndex)

    <sempredFuncs.values; separator="\n">
<endif>



>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
def action(self, localctx, ruleIndex, actionIndex):
    if self._actions is None:
        actions = dict()
<recog.actionFuncs.values:{f|
        actions[<f.ruleIndex>] = self.<f.name>_action }; separator="\n">
        self._actions = actions
    action = self._actions.get(ruleIndex, None)
    if action is not None:
        action(localctx, actionIndex)
    else:
        raise Exception("No registered action for:" + str(ruleIndex))

<actionFuncs.values; separator="\n">

<endif>
<if(sempredFuncs)>
def sempred(self, localctx, ruleIndex, predIndex):
    if self._predicates is None:
        preds = dict()
<recog.sempredFuncs.values:{f|
        preds[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n">
        self._predicates = preds
    pred = self._predicates.get(ruleIndex, None)
    if pred is not None:
        return pred(localctx, predIndex)
    else:
        raise Exception("No registered predicate for:" + str(ruleIndex))

<sempredFuncs.values; separator="\n">
<endif>
>>

parser_ctor(p) ::= <<
def __init__(self, input, output=sys.stdout):
    super(<parser.name>, self).__init__(input, output=output)
```

```
    self.checkVersion("<file.ANTLRVersion>")
    self._interp = ParserATNSimulator(self, self.atn, self.decisionsToDFA, self.sharedContextCache)
    self._predicates = None

>>

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<

def <r.name>_action(self, localctx , actionIndex):
<actions:{index|
<if(first(actions))>
    if actionIndex == <index>:
        <actions.(index)>
<elseif(rest(actions))>
    elif actionIndex == <index>:
        <actions.(index)>
<endif> }; separator="\n">
>>

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
def <r.name>_sempred(self, localctx, predIndex):
    <actions:{index|
<if(first(actions))>
    if predIndex == <index>:
        return <actions.(index)>
<elseif(rest(actions))>
    elif predIndex == <index>:
        return <actions.(index)>
<endif> }; separator="\n">

>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<ruleCtx>

<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

def <currentRule.name>(self<currentRule.args:{a | , <a.name>}>):

    localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, self.state<currentRule.args:{a | , <a.name>}>)
```

```
  self.enterRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>)
  <namedActions.init>
  <locals; separator="\n">
  try:
     <code>
     <postamble; separator="\n">
     <namedActions.after>
  <if(exceptions)>
  <exceptions; separator="\n">
  <else>
  except RecognitionException as re:
     localctx.exception = re
     self._errHandler.reportError(self, re)
     self._errHandler.recover(self, re)
  <endif>
  finally:
     <finallyAction>
     self.exitRule()
  return localctx

>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
 namedActions,finallyAction,postamble) ::=
<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

def <currentRule.name>(self, _p=0<if(currentRule.args)>, <args:{a | , <a>}><endif>):
  _parentctx = self._ctx
  _parentState = self.state
  localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, _parentState<args:{a | , <a.name>}>)
  _prevctx = localctx
  _startState = <currentRule.startState>
  self.enterRecursionRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>, _p)
  <namedActions.init>
  <locals; separator="\n">
  try:
     <code>
     <postamble; separator="\n">
     <namedActions.after>
  except RecognitionException as re:
     localctx.exception = re
     self._errHandler.reportError(self, re)
     self._errHandler.recover(self, re)
  finally:
     <finallyAction>
```

```
      self.unrollRecursionContexts(_parentctx)
  return localctx

>>

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>localctx = <parser.name>.<currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(self, localctx)<endif>
self.enterOuterAlt(localctx, <currentOuterMostAltCodeBlock.alt.altNum>)
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>

LL1AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  pass}; separator="\nel">
else:
  <error>

>>

LL1OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  pass}; separator="\nel">
else:
  pass
>>

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
if <expr>:
```

```
  <alts; separator="\n">

<!else if ( !(<followExpr>) ) <error>!>
>>


LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
while <loopExpr>:
  <alts; separator="\n">
  self.state = <choice.loopBackStateNumber>
  self._errHandler.sync(self)
  <iteration>

>>

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
<preamble; separator="\n">
while True:
  <alts; separator="\n">
  self.state = <choice.stateNumber> <! loopback/exit decision !>
  self._errHandler.sync(self)
  <iteration>
  if not (<loopExpr>):
     break

>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{alt |
if la_ == <i>:
  <alt>
  pass
}; separator="\nel">

>>
```

```
OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{alt |
if la_ == <i><if(!choice.ast.greedy)>+1<endif>:
  <alt>
}; separator="\nel">

>>

StarBlock(choice, alts, sync, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
_alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
  if _alt==1<if(!choice.ast.greedy)>+1<endif>:
    <iteration>
    <alts> <! should only be one !>
  self.state = <choice.loopBackStateNumber>
  self._errHandler.sync(self)
  _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)

>>

PlusBlock(choice, alts, error) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
_alt = 1<if(!choice.ast.greedy)>+1<endif>
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
  <alts:{alt|
if _alt == <i><if(!choice.ast.greedy)>+1<endif>:
  <alt>
}; separator="\nel">
  else:
    <error>
  self.state = <choice.loopBackStateNumber> <! loopback/exit decision !>
  self._errHandler.sync(self)
  _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)

>>

Sync(s) ::= "sync(<s.expecting.name>)"

ThrowNoViableAlt(t) ::= "raise NoViableAltException(self)"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
```

```
bits)><endif>}; separator=" or ">
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShiftVar(s.varName, bits.shift)>})> and ((1 \<\< <offsetShiftVar(s.varName,
bits.shift)>) & (<bits.ttypes:{ttype | (1 \<\< <offsetShiftType(ttype, bits.shift)>)}; separator=" | ">)) != 0)
%>

isZero ::= [
"0":true,
default:false
]

offsetShiftVar(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(<shiftAmount> - <offset>)<else><shiftAmount><endif>
%>

offsetShiftType(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(<parser.name>.<shiftAmount> - <offset>)<else><parser.name>.<shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | <s.varName>==<parser.name>.<ttype>}; separator=" or ">
%>

cases(ttypes) ::= <<
if token in [<ttypes:{t | <parser.name>.<t>}; separator=", ">]:
>>

InvokeRule(r, argExprsChunks) ::= <<
self.state = <r.stateNumber>
<if(r.labels)><r.labels:{l | <labelref(l)> =
}><endif>self.<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChu
nks>)
>>

MatchToken(m) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>self.match(<parser.name>.<m.name>)
>>
```

```
MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels:{l | <labelref(l)> = }>self._input.LT(1)<endif>
<capture>
<if(invert)>if <m.varName> \<= 0 or <expr><else>if not(<expr>)<endif>:
<if(m.labels)><m.labels:{l |     <labelref(l)> = }><else>    <endif>self._errHandler.recoverInline(self)
else:
  self._errHandler.reportMatch(self)
  self.consume()
>>

Wildcard(w) ::= <<
self.state = <w.stateNumber>
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>self.matchWildcard()
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
self.state = <p.stateNumber>
if not <chunks>:
  from antlr4.error.Errors import FailedPredicateException
  raise FailedPredicateException(self, <p.predicate><if(failChunks)>, <failChunks><elseif(p.msg)>,
<p.msg><endif>)
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
except <catchArg>:
  <catchAction>
>>

// lexer actions are not associated with model objects

LexerSkipCommand()  ::= "skip()"
LexerMoreCommand()  ::= "more()"
LexerPopModeCommand() ::= "popMode()"

LexerTypeCommand(arg, grammar)      ::= "_type = <arg>"
LexerChannelCommand(arg, grammar)   ::= "_channel = <arg>"
LexerModeCommand(arg, grammar)      ::= "_mode = <arg>"
```

```
LexerPushModeCommand(arg, grammar)  ::= "pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "localctx.<a.name>"
LocalRef(a) ::= "localctx.<a.name>"
RetValueRef(a) ::= "localctx.<a.name>"
QRetValueRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(None if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.text)"
TokenPropertyRef_type(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.type)"
TokenPropertyRef_line(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.line)"
TokenPropertyRef_pos(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.column)"
TokenPropertyRef_channel(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.channel)"
TokenPropertyRef_index(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.tokenIndex)"
TokenPropertyRef_int(t) ::= "(0 if <ctx(t)>.<t.label> is None else int(<ctx(t)>.<t.label>.text))"

RulePropertyRef_start(r) ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.start)"
RulePropertyRef_stop(r)  ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.stop)"
RulePropertyRef_text(r)  ::= "(None if <ctx(r)>.<r.label> is None else
self._input.getText(<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop))"
RulePropertyRef_ctx(r)  ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r)  ::= "self"

ThisRulePropertyRef_start(r) ::= "localctx.start"
ThisRulePropertyRef_stop(r)  ::= "localctx.stop"
ThisRulePropertyRef_text(r)  ::= "self._input.getText(localctx.start, self._input.LT(-1))"
ThisRulePropertyRef_ctx(r)  ::= "localctx"
ThisRulePropertyRef_parser(r)  ::= "self"

NonLocalAttrRef(s)     ::= "self.getInvokingContext(<s.ruleIndex>).<s.name>"
SetNonLocalAttr(s, rhsChunks)  ::= "self.getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.append(<labelref(a.label)>)"

TokenDecl(t) ::= "self.<t.name> = None # <TokenLabelType()>"
TokenTypeDecl(t) ::= "self.<t.name> = 0 # <TokenLabelType()> type"
TokenListDecl(t) ::= "self.<t.name> = list() # of <TokenLabelType()>s"
RuleContextDecl(r) ::= "self.<r.name> = None # <r.ctxName>"
RuleContextListDecl(rdecl) ::= "self.<rdecl.name> = list() # of <rdecl.ctxName>s"
```

```
ContextTokenGetterDecl(t)      ::= <<
def <t.name>(self):
  return self.getToken(<parser.name>.<t.name>, 0)
>>

// should never be called
ContextTokenListGetterDecl(t)  ::= <<
def <t.name>_list(self):
  return self.getTokens(<parser.name>.<t.name>)
>>

ContextTokenListIndexedGetterDecl(t)  ::= <<
def <t.name>(self, i=None):
  if i is None:
     return self.getTokens(<parser.name>.<t.name>)
  else:
     return self.getToken(<parser.name>.<t.name>, i)
>>

ContextRuleGetterDecl(r)       ::= <<
def <r.name>(self):
  return self.getTypedRuleContext(<parser.name>.<r.ctxName>,0)

>>

// should never be called
ContextRuleListGetterDecl(r)   ::= <<
def <r.name>_list(self):
  return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)

>>

ContextRuleListIndexedGetterDecl(r)   ::= <<
def <r.name>(self, i=None):
  if i is None:
     return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)
  else:
     return self.getTypedRuleContext(<parser.name>.<r.ctxName>,i)

>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
*  r becomes rContext.
*/
RuleContextNameSuffix() ::= "Context"
```

```
ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName)   ::= "_<ruleName>"
ImplicitSetLabel(id)    ::= "_tset<id>"
ListLabelName(label)    ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = self._input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = self._input.LA(1)"

StructDecl(struct,ctorAttrs,attrs,getters,dispatchMethods,interfaces,extensionMembers) ::= <<
class <struct.name>(<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif>):

  def __init__(self, parser, parent=None, invokingState=-1<struct.ctorAttrs:{a | , <a.name>=None}>):
    super(<parser.name>.<struct.name>, self).__init__(parent, invokingState)
    self.parser = parser
    <attrs:{a | <a>}; separator="\n">
    <struct.ctorAttrs:{a | self.<a.name> = <a.name>}; separator="\n">

  <getters:{g | <g>}; separator="\n\n">

  def getRuleIndex(self):
    return <parser.name>.RULE_<struct.derivedFromName>

<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
  def copyFrom(self, ctx):
    super(<parser.name>.<struct.name>, self).copyFrom(ctx)
    <struct.attrs:{a | self.<a.name> = ctx.<a.name>}; separator="\n">

<endif>
  <dispatchMethods; separator="\n">
  <extensionMembers; separator="\n">

>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
class <struct.name>(<currentRule.name; format="cap">Context):

  def __init__(self, parser, ctx): # actually a <parser.name>.<currentRule.name; format="cap">Context)
    super(<parser.name>.<struct.name>, self).__init__(parser)
    <attrs:{a | <a>}; separator="\n">
    self.copyFrom(ctx)

  <getters:{g | <g>}; separator="\n">

  <dispatchMethods; separator="\n">

>>
```

```
ListenerDispatchMethod(method) ::= <<
def <if(method.isEnter)>enter<else>exit<endif>Rule(self, listener):
  if hasattr(listener, "<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">"):
    listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(self)

>>

VisitorDispatchMethod(method) ::= <<
def accept(self, visitor):
  if hasattr(visitor, "visit<struct.derivedFromName; format="cap">"):
    return visitor.visit<struct.derivedFromName; format="cap">(self)
  else:
    return visitor.visitChildren(self)

>>

AttributeDecl(d) ::= "self.<d.name> = <if(d.initValue)><d.initValue><else>None<endif>"

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>localctx.<endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "localctx"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec)  ::= "self.precpred(self._ctx, <opPrec>)"
recRuleSetReturnAction(src,name)   ::= "$<name>=$<src>.<name>"
recRuleSetStopToken()                 ::= "self._ctx.stop = self._input.LT(-1)"

recRuleAltStartAction(ruleName, ctxName, label) ::= <<
localctx = <parser.name>.<ctxName>Context(self, _parentctx, _parentState)
<if(label)>localctx.<label> = _prevctx<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
localctx = <parser.name>.<currentAltLabel; format="cap">Context(self, <parser.name>.<ruleName;
format="cap">Context(self, _parentctx, _parentState))
<if(label)>
<if(isListLabel)>
localctx.<label>.append(_prevctx)
<else>
localctx.<label> = _prevctx
<endif>
<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>
```

```
recRuleReplaceContext(ctxName) ::= <<
localctx = <parser.name>.<ctxName>Context(self, localctx)
self._ctx = localctx
_prevctx = localctx
>>

recRuleSetPrevCtx() ::= <<
if self._parseListeners is not None:
  self.triggerExitRuleEvent()
_prevctx = localctx
>>


LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
# encoding: utf-8
from __future__ import print_function
from antlr4 import *
from io import StringIO
import sys

<namedActions.header>

<lexer>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
<if(superClass)>
if __name__ is not None and "." in __name__:
  from .<superClass> import <superClass>
else:
  from <superClass> import <superClass>

<endif>

<atn>

class <lexer.name>(<if(superClass)><superClass><else>Lexer<endif>):

  atn = ATNDeserializer().deserialize(serializedATN())

  decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]

<if(lexer.channels)>
  <lexer.channels:{c| <c> = <lexer.channels.(c)>}; separator="\n">

<endif>
```

```
<if(rest(lexer.modes))>
  <rest(lexer.modes):{m| <m> = <i>}; separator="\n">

<endif>
  <if(lexer.tokens)>
  <lexer.tokens:{k | <k> = <lexer.tokens.(k)>}; separator="\n", wrap, anchor>
  <endif>

  channelNames = [ u"DEFAULT_TOKEN_CHANNEL", u"HIDDEN"<if (lexer.channels)>, <lexer.channels:{c|
u"<c>"}; separator=", ", wrap, anchor><endif> ]

  modeNames = [ <lexer.modes:{m| u"<m>"}; separator=", ", wrap, anchor> ]

  literalNames = [ u"\<INVALID>",
        <lexer.literalNames:{t | u<t>}; separator=", ", wrap, anchor> ]

  symbolicNames = [ u"\<INVALID>",
        <lexer.symbolicNames:{t | u<t>}; separator=", ", wrap, anchor> ]

  ruleNames = [ <lexer.ruleNames:{r | u"<r>"}; separator=", ", wrap, anchor> ]

  grammarFileName = u"<lexer.grammarFileName>"

  def __init__(self, input=None, output=sys.stdout):
     super(<lexer.name>, self).__init__(input, output=output)
     self.checkVersion("<lexerFile.ANTLRVersion>")
     self._interp = LexerATNSimulator(self, self.atn, self.decisionsToDFA, PredictionContextCache())
     self._actions = None
     self._predicates = None

  <namedActions.members>

  <dumpActions(lexer, "", actionFuncs, sempredFuncs)>

>>

SerializedATN(model) ::= <<
<! only one segment, can be inlined !>

def serializedATN():
  with StringIO() as buf:
     buf.write(u"<model.serialized; wrap={")<\n>       buf.write(u"}>")
     return buf.getvalue()

>>

/** Using a type to init value map, try to init a type; if not in table
*  must be an object, default value is "null".
```

```
*/
initValue(typeName) ::= <<
<pythonTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".py"
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/codegen/Python2/Python2.stg
No license file was found, but licenses were detected in source scan.


```
/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
```


```
tree grammar SourceGenTriggers;
options {
 language     = Java;
 tokenVocab   = ANTLRParser;
 ASTLabelType = GrammarAST;
}
```

```
@header {
package org.antlr.v4.codegen;
import org.antlr.v4.misc.Utils;
import org.antlr.v4.codegen.model.*;
import org.antlr.v4.codegen.model.decl.*;
import org.antlr.v4.tool.*;
import org.antlr.v4.tool.ast.*;
import java.util.Collections;
import java.util.Map;
import java.util.HashMap;
}

@members {
 public OutputModelController controller;
  public boolean hasLookaheadBlock;
  public SourceGenTriggers(TreeNodeStream input, OutputModelController controller) {
   this(input);
   this.controller = controller;
  }
}

dummy : block[null, null] ;

block[GrammarAST label, GrammarAST ebnfRoot] returns [List<? extends SrcOp> omos]
  : ^( blk=BLOCK (^(OPTIONS .+))?
  {List<CodeBlockForAlt> alts = new ArrayList<CodeBlockForAlt>();}
    ( alternative {alts.add($alternative.altCodeBlock);} )+
   )
   {
   if ( alts.size()==1 && ebnfRoot==null) return alts;
   if ( ebnfRoot==null ) {
      $omos = DefaultOutputModelFactory.list(controller.getChoiceBlock((BlockAST)$blk, alts, $label));
   }
   else {
        Choice choice = controller.getEBNFBlock($ebnfRoot, alts);
        hasLookaheadBlock |= choice instanceof PlusBlock || choice instanceof StarBlock;
      $omos = DefaultOutputModelFactory.list(choice);
   }
   }
  ;

alternative returns [CodeBlockForAlt altCodeBlock, List<SrcOp> ops]
@init {
  boolean outerMost = inContext("RULE BLOCK");
}
@after {
  controller.finishAlternative($altCodeBlock, $ops, outerMost);
```

```
}
  : a=alt[outerMost] {$altCodeBlock=$a.altCodeBlock; $ops=$a.ops;}
 ;

alt[boolean outerMost] returns [CodeBlockForAlt altCodeBlock, List<SrcOp> ops]
@init {
 // set alt if outer ALT only (the only ones with alt field set to Alternative object)
 AltAST altAST = (AltAST)retval.start;
 if ( outerMost ) controller.setCurrentOuterMostAlt(altAST.alt);
}
 : {
  List<SrcOp> elems = new ArrayList<SrcOp>();
  // TODO: shouldn't we pass $start to controller.alternative()?
  $altCodeBlock = controller.alternative(controller.getCurrentOuterMostAlt(), outerMost);
  $altCodeBlock.ops = $ops = elems;
  controller.setCurrentBlock($altCodeBlock);
  }
  ^( ALT elementOptions? ( element {if ($element.omos!=null) elems.addAll($element.omos);} )+ )

 | ^(ALT elementOptions? EPSILON)
      {$altCodeBlock = controller.epsilon(controller.getCurrentOuterMostAlt(), outerMost);}
  ;

element returns [List<? extends SrcOp> omos]
 : labeledElement     {$omos = $labeledElement.omos;}
 | atom[null,false]   {$omos = $atom.omos;}
 | subrule       {$omos = $subrule.omos;}
 |   ACTION       {$omos = controller.action((ActionAST)$ACTION);}
 |   SEMPRED       {$omos = controller.sempred((ActionAST)$SEMPRED);}
 | ^(ACTION elementOptions)  {$omos = controller.action((ActionAST)$ACTION);}
 |   ^(SEMPRED elementOptions)  {$omos = controller.sempred((ActionAST)$SEMPRED);}
 ;

labeledElement returns [List<? extends SrcOp> omos]
 : ^(ASSIGN ID atom[$ID,false] )   {$omos = $atom.omos;}
 | ^(PLUS_ASSIGN ID atom[$ID,false])  {$omos = $atom.omos;}
 | ^(ASSIGN ID block[$ID,null] )   {$omos = $block.omos;}
 | ^(PLUS_ASSIGN ID block[$ID,null])  {$omos = $block.omos;}
 ;

subrule returns [List<? extends SrcOp> omos]
 : ^(OPTIONAL b=block[null,$OPTIONAL])
  {
  $omos = $block.omos;
  }
 | ( ^(op=CLOSURE b=block[null,null])
  | ^(op=POSITIVE_CLOSURE b=block[null,null])
  )
```

```
 {
 List<CodeBlockForAlt> alts = new ArrayList<CodeBlockForAlt>();
 SrcOp blk = $b.omos.get(0);
 CodeBlockForAlt alt = new CodeBlockForAlt(controller.delegate);
 alt.addOp(blk);
 alts.add(alt);
 SrcOp loop = controller.getEBNFBlock($op, alts); // "star it"
     hasLookaheadBlock |= loop instanceof PlusBlock || loop instanceof StarBlock;
     $omos = DefaultOutputModelFactory.list(loop);
 }
| block[null, null]     {$omos = $block.omos;}
  ;

blockSet[GrammarAST label, boolean invert] returns [List<SrcOp> omos]
  : ^(SET atom[label,invert]+) {$omos = controller.set($SET, $label, invert);}
  ;

/*
setElement
 : STRING_LITERAL
 | TOKEN_REF
 | ^(RANGE STRING_LITERAL STRING_LITERAL)
 ;
*/

// TODO: combine ROOT/BANG into one then just make new op ref'ing return value of atom/terminal...
// TODO: same for NOT
atom[GrammarAST label, boolean invert] returns [List<SrcOp> omos]
 : ^(NOT a=atom[$label, true])  {$omos = $a.omos;}
 | range[label]       {$omos = $range.omos;}
 | ^(DOT ID terminal[$label])
 | ^(DOT ID ruleref[$label])
  | ^(WILDCARD .)       {$omos = controller.wildcard($WILDCARD, $label);}
  | WILDCARD        {$omos = controller.wildcard($WILDCARD, $label);}
  |   terminal[label]     {$omos = $terminal.omos;}
  |   ruleref[label]     {$omos = $ruleref.omos;}
 | blockSet[$label, invert]  {$omos = $blockSet.omos;}
 ;

ruleref[GrammarAST label] returns [List<SrcOp> omos]
  : ^(RULE_REF ARG_ACTION? elementOptions?)  {$omos = controller.ruleRef($RULE_REF, $label,
$ARG_ACTION);}
  ;

range[GrammarAST label] returns [List<SrcOp> omos]
  : ^(RANGE a=STRING_LITERAL b=STRING_LITERAL)
  ;
```

```
terminal[GrammarAST label] returns [List<SrcOp> omos]
  :  ^(STRING_LITERAL .)   {$omos = controller.stringRef($STRING_LITERAL, $label);}
  | STRING_LITERAL    {$omos = controller.stringRef($STRING_LITERAL, $label);}
  | ^(TOKEN_REF ARG_ACTION .) {$omos = controller.tokenRef($TOKEN_REF, $label, $ARG_ACTION);}
  | ^(TOKEN_REF .)    {$omos = controller.tokenRef($TOKEN_REF, $label, null);}
  | TOKEN_REF     {$omos = controller.tokenRef($TOKEN_REF, $label, null);}
  ;

elementOptions
  : ^(ELEMENT_OPTIONS elementOption+)
  ;

elementOption
  : ID
  |  ^(ASSIGN ID ID)
  |  ^(ASSIGN ID STRING_LITERAL)
  |  ^(ASSIGN ID ACTION)
  |  ^(ASSIGN ID INT)
  ;
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/SourceGenTriggers.g
No license file was found, but licenses were detected in source scan.

/*
[The "BSD license"]
Copyright (c) 2010 Terence Parr
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ATNBuilder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.
* Use of this file is governed by the BSD 3-clause license that
* can be found in the LICENSE.txt file in the project root.
*/

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/AttributeResolver.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/target/JavaScriptTarget.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/analysis/LeftRecursionDetector.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/ArgRef.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/RuleRefAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/TokenDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_int.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/SemPred.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/RuleElementAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ToolMessage.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/ThisRulePropertyRef_parser.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_index.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/target/SwiftTarget.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/GrammarASTWithOptions.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/LabelType.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ScopeParser.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/semantics/SymbolCollector.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/ContextRuleListIndexedGetterDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/RuleFunction.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-

jar/org/antlr/v4/semantics/RuleCollector.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/LabeledOp.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/Wildcard.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/automata/LexerATNFactory.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/ContextRuleListGetterDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/VisitorDispatchMethod.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/RangeAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/v3TreeGrammarException.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/OptionalBlock.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/ThrowEarlyExitException.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/analysis/LeftRecursiveRuleTransformer.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ErrorManager.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/AltLabelStructDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/GrammarInterpreterRuleContext.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/ActionText.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/BuildDependencyGenerator.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/automata/ATNVisitor.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/RuleContextDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/NonLocalAttrRef.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/PostScriptDocument.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/OutputModelFactory.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ANTLRToolListener.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/CodeGenPipeline.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/RuleElement.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/Alternative.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/Target.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/AltBlock.java

* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/LL1Choice.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/automata/TailEpsilonRemover.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/semantics/BlankActionSplitterListener.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/analysis/LeftRecursiveRuleAltInfo.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/AttributeDict.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/target/Python2Target.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/ArgAction.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_line.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef_stop.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef_text.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/PredAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/GrammarSyntaxMessage.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/OutputModelWalker.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/GrammarASTVisitor.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/DefaultToolListener.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/ActionTemplate.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/ElementListDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/Choice.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/ListLabelRef.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/UnicodeEscapes.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/RuleActionFunction.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/MatchToken.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/LocalRef.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/ThisRulePropertyRef_ctx.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/CaptureNextTokenType.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-

jar/org/antlr/v4/codegen/model/OutputFile.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/LL1OptionalBlockSingleAlt.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/ThisRulePropertyRef_stop.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/AddToLabelList.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/AltAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/MatchSet.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ToolANTLRLexer.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/GrammarToken.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/BaseListenerFile.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/TreeLayoutAdaptor.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/SystemFontMetrics.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/Tool.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/ModelElement.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/BaseVisitorFile.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/target/CSharpTarget.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef_ctx.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ResyncToEndOfRuleBlock.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/GrammarTransformPipeline.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/RuleAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/misc/EscapeSequenceParsing.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/LabelElementPair.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_pos.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/ParserFile.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/automata/ATNOptimizer.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/semantics/SymbolChecks.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/MatchNotSet.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-

jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_type.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/ContextTokenListGetterDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/BasicFontMetrics.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/ParserFactory.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ToolANTLRParser.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ErrorType.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/QuantifierAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/Lexer.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/LL1Loop.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/BlankOutputModelFactory.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/misc/MutableInt.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/ThrowRecognitionException.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/target/JavaTarget.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/GrammarParserInterpreter.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/ThisRulePropertyRef_text.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_channel.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/Decl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/SerializedATN.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/LabelRef.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/DOTGenerator.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/automata/ATNFactory.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/ActionAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/TreePostScriptGenerator.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/NotAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/JFileChooserConfirmOverwrite.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/ExceptionClause.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/semantics/BasicSemanticChecks.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/LexerFactory.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/RuleSempredFunction.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/CodeBlockForOuterMostAlt.java

* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/InvokeRule.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/TokenVocabParser.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/OutputModelObject.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/CodeBlockForAlt.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/semantics/ActionSniffer.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/ContextTokenListIndexedGetterDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ActionSplitterListener.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/Loop.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/PlusBlock.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/ContextGetterDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/Attribute.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/ThisRulePropertyRef_start.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/v4ParserException.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/analysis/AnalysisPipeline.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/StructDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/QRetValueRef.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/SrcOp.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/Rule.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/semantics/UseDefAnalyzer.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/TokenListDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/unicode/UnicodeDataTemplateController.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/ListenerDispatchMethod.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/SetNonLocalAttr.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/semantics/SemanticPipeline.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/TreeViewer.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/CodeGeneratorExtension.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-

jar/org/antlr/v4/codegen/model/CaptureNextToken.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/ThrowNoViableAlt.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ErrorSeverity.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/tool/LeftRecursionCyclesMessage.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/DispatchMethod.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/target/Python3Target.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/LexerFile.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/LL1AltBlock.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef_parser.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/dbg.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/Recognizer.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/target/DartTarget.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/SetAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/LeftRecursiveRule.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/chunk/RetValueRef.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/TestRig.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/Action.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/DefaultOutputModelFactory.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/Sync.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/misc/CharSupport.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/decl/ContextRuleGetterDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/OutputModelController.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/automata/ATNPrinter.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/LeftRecursiveRuleFunction.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/ListenerFile.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/StarBlockAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/misc/Graph.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/decl/TokenTypeDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/semantics/AttributeChecks.java

* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/ElementFrequenciesVisitor.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/RuleContextListDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/OptionalBlockAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/GrammarSemanticsMessage.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/ActionChunk.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/ActionTranslator.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_text.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/CodeGenerator.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/analysis/LeftRecursiveRuleAnalyzer.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/LL1PlusBlockSingleAlt.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/VisitorFile.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/PlusBlockAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/GrammarRootAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/GrammarASTAdaptor.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/Trees.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/gui/TreeTextProvider.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/target/CppTarget.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/SetAttr.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/TerminalAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/misc/Utils.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/target/GoTarget.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/CodeBlock.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/StarBlock.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/ContextTokenGetterDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/decl/AttributeDecl.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef_start.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-

jar/org/antlr/v4/automata/ParserATNFactory.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/codegen/model/Parser.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/chunk/TokenRef.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/misc/OrderedHashMap.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ANTLRMessage.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/target/PHPTarget.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/misc/FrequencySet.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/LexerGrammar.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/Grammar.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/BlockAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/tool/ast/GrammarASTErrorNode.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/LL1StarBlockSingleAlt.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/LL1OptionalBlock.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/ast/GrammarAST.java
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-
jar/org/antlr/v4/codegen/model/TestSetInline.java
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
*   derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

```
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

tree grammar ATNBuilder;
options {
 language     = Java;
 tokenVocab   = ANTLRParser;
 ASTLabelType = GrammarAST;
// filter       = true;
}

// Include the copyright in this source and also the generated source
@header {
/*
[The "BSD license"]
Copyright (c) 2010 Terence Parr
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
package org.antlr.v4.parse;
import org.antlr.v4.tool.*;
import org.antlr.v4.tool.ast.*;
import org.antlr.v4.automata.ATNFactory;
}

@members {
  ATNFactory factory;
```

```
  public ATNBuilder(TreeNodeStream input, ATNFactory factory) {
   this(input);
   this.factory = factory;
  }
}

dummy : block[null] ; // avoid error about no start rule

ruleBlock[GrammarAST ebnfRoot] returns [ATNFactory.Handle p]
@init {
  List<ATNFactory.Handle> alts = new ArrayList<ATNFactory.Handle>();
  int alt = 1;
  factory.setCurrentOuterAlt(alt);
}
  : ^(BLOCK
        (^(OPTIONS .*))?
        (   a=alternative
           {alts.add($a.p); factory.setCurrentOuterAlt(++alt);}
        )+
     )
   {$p = factory.block((BlockAST)$BLOCK, ebnfRoot, alts);}
  ;

block[GrammarAST ebnfRoot] returns [ATNFactory.Handle p]
@init {List<ATNFactory.Handle> alts = new ArrayList<ATNFactory.Handle>();}
  : ^(BLOCK (^(OPTIONS .*))? (a=alternative {alts.add($a.p);})+)
   {$p = factory.block((BlockAST)$BLOCK, ebnfRoot, alts);}
  ;

alternative returns [ATNFactory.Handle p]
@init {List<ATNFactory.Handle> els = new ArrayList<ATNFactory.Handle>();}
  : ^(LEXER_ALT_ACTION a=alternative lexerCommands)
     {$p = factory.lexerAltCommands($a.p,$lexerCommands.p);}
  | ^(ALT elementOptions? EPSILON)       {$p = factory.epsilon($EPSILON);}
  |   ^(ALT elementOptions? (e=element {els.add($e.p);})+) {$p = factory.alt(els);}
  ;

lexerCommands returns [ATNFactory.Handle p]
@init {List<ATNFactory.Handle> cmds = new ArrayList<ATNFactory.Handle>();}
  :   (c=lexerCommand {if ($c.cmd != null) cmds.add($c.cmd);})+
     {
     $p = factory.alt(cmds);
     }
  ;

lexerCommand returns [ATNFactory.Handle cmd]
 : ^(LEXER_ACTION_CALL ID lexerCommandExpr)
     {$cmd = factory.lexerCallCommand($ID, $lexerCommandExpr.start);}
```

```
 | ID
       {$cmd = factory.lexerCommand($ID);}
 ;

lexerCommandExpr
 : ID
 | INT
 ;

element returns [ATNFactory.Handle p]
 : labeledElement    {$p = $labeledElement.p;}
 | atom      {$p = $atom.p;}
 | subrule      {$p = $subrule.p;}
 |   ACTION      {$p = factory.action((ActionAST)$ACTION);}
 |   SEMPRED      {$p = factory.sempred((PredAST)$SEMPRED);}
 |   ^(ACTION .)     {$p = factory.action((ActionAST)$ACTION);}
 |   ^(SEMPRED .)    {$p = factory.sempred((PredAST)$SEMPRED);}
   | ^(NOT b=blockSet[true])  {$p = $b.p;}
   | LEXER_CHAR_SET    {$p = factory.charSetLiteral($start);}
 ;

astOperand returns [ATNFactory.Handle p]
 : atom      {$p = $atom.p;}
 | ^(NOT blockSet[true])  {$p = $blockSet.p;}
 ;

labeledElement returns [ATNFactory.Handle p]
 : ^(ASSIGN ID element)     {$p = factory.label($element.p);}
 | ^(PLUS_ASSIGN ID element)   {$p = factory.listLabel($element.p);}
 ;

subrule returns [ATNFactory.Handle p]
 : ^(OPTIONAL block[$start])   {$p = $block.p;}
 | ^(CLOSURE block[$start])   {$p = $block.p;}
 | ^(POSITIVE_CLOSURE block[$start]) {$p = $block.p;}
 |  block[null]       {$p = $block.p;}
   ;

blockSet[boolean invert] returns [ATNFactory.Handle p]
@init {List<GrammarAST> alts = new ArrayList<GrammarAST>();}
 : ^(SET (setElement {alts.add($setElement.start);})+) {$p = factory.set($start, alts, $invert);}
 ;

/** Don't combine with atom otherwise it will build spurious ATN nodes */
setElement
 : ^(STRING_LITERAL .)
 | ^(TOKEN_REF .)
 | STRING_LITERAL
```

```
 | TOKEN_REF
 | ^(RANGE a=STRING_LITERAL b=STRING_LITERAL)
   |   LEXER_CHAR_SET
 ;

atom returns [ATNFactory.Handle p]
 : range     {$p = $range.p;}
 | ^(DOT ID terminal)  {$p = $terminal.p;}
 | ^(DOT ID ruleref)  {$p = $ruleref.p;}
   | ^(WILDCARD .)   {$p = factory.wildcard($start);}
   | WILDCARD    {$p = factory.wildcard($start);}
   | blockSet[false]   {$p = $blockSet.p;}
   |   terminal    {$p = $terminal.p;}
   |   ruleref     {$p = $ruleref.p;}
   ;

ruleref returns [ATNFactory.Handle p]
   : ^(RULE_REF ARG_ACTION? ^(ELEMENT_OPTIONS .*))  {$p = factory.ruleRef($RULE_REF);}
   | ^(RULE_REF ARG_ACTION?)        {$p = factory.ruleRef($RULE_REF);}
   | RULE_REF          {$p = factory.ruleRef($RULE_REF);}
   ;

range returns [ATNFactory.Handle p]
   : ^(RANGE a=STRING_LITERAL b=STRING_LITERAL) {$p = factory.range($a,$b);}
   ;

terminal returns [ATNFactory.Handle p]
   :  ^(STRING_LITERAL .)   {$p = factory.stringLiteral((TerminalAST)$start);}
   | STRING_LITERAL    {$p = factory.stringLiteral((TerminalAST)$start);}
   | ^(TOKEN_REF ARG_ACTION .) {$p = factory.tokenRef((TerminalAST)$start);}
   | ^(TOKEN_REF .)    {$p = factory.tokenRef((TerminalAST)$start);}
   | TOKEN_REF     {$p = factory.tokenRef((TerminalAST)$start);}
   ;

elementOptions
 : ^(ELEMENT_OPTIONS elementOption*)
 ;

elementOption
 : ID
 | ^(ASSIGN ID ID)
 | ^(ASSIGN ID STRING_LITERAL)
 | ^(ASSIGN ID ACTION)
 | ^(ASSIGN ID INT)
 ;

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ATNBuilder.g
```

No license file was found, but licenses were detected in source scan.

```
/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

atn(startState, states, edges, rankdir, decisionRanks, useBox) ::= <<
digraph ATN {
rankdir=LR;
<decisionRanks; separator="\n">
<states; separator="\n">
<edges; separator="\n">
}
>>

dfa(name, startState, states, edges, rankdir, decisionRanks, useBox) ::= <<
digraph <name>  {
<if(rankdir)>rankdir=<rankdir>;<endif>
<decisionRanks; separator="\n">
<states; separator="\n">
<edges; separator="\n">
}
```

```
>>

decision-rank(states) ::= <<
{rank=same; rankdir=TB; <states:{s | s<s>}; separator="; ">}
>>

edge(src,target,label,arrowhead,transitionIndex) ::= <<
<src><if(transitionIndex)>:p<transitionIndex><endif> -> <target> [fontsize=11, fontname="Courier", arrowsize=.7,
label = "<label>"<if(arrowhead)>, arrowhead = <arrowhead><endif>];
>>

action-edge(src,target,label,arrowhead,transitionIndex) ::= <<
<src><if(transitionIndex)>:p<transitionIndex><endif> -> <target> [fontsize=11, fontname="Courier", arrowsize=.7,
label = "<label>"<if(arrowhead)>, arrowhead = <arrowhead><endif>];
>>

epsilon-edge(src,label,target,arrowhead,transitionIndex,loopback=false) ::= <<
<src><if(transitionIndex)>:p<transitionIndex><endif> -> <target> [fontname="Times-Italic",
label="&epsilon;"<if(loopback)>, style="dashed"<endif>];
>>

state(state, label, name, transitions) ::= <%
<name>[fontsize=11,
  label="
    <! rest(transition) tests for decision states: these nodes have a non-empty set of transitions after the first one. !>
    <if(rest(transitions))>
      {
      <! Label on the left side of the record node. !>
      <label>
      |
      <! Named ports in order on right side of record node, no display text. !>
      {<transitions:{t|\<p<i0>>}; separator="|">}}
    <else>
      <label>
    <endif>
    "
  <if(rest(transitions))>
    , shape=record, fixedsize=false
  <else>
    , shape=circle, fixedsize=true, width=.55
  <endif>
  , peripheries=1];
%>

stopstate(name,label,actionIndex,useBox) ::= <<
<name>[fontsize=11, label="<label><if(actionIndex)>,\naction:<actionIndex><endif>",
<if(useBox)>shape=polygon,sides=4,peripheries=2,fixedsize=false<else>shape=doublecircle, fixedsize=true,
width=.6<endif>];
```

>>

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/dot/graphs.stg
No license file was found, but licenses were detected in source scan.

```
/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

/** templates used to generate make-compatible dependencies */

/** Generate "f : x, y, z" dependencies for input
*  dependencies and generated files. in and out
*  are File objects.  For example, you can say
*  <f.canonicalPath>
*/
dependencies(grammarFileName,in,out) ::= <<
<if(in)><grammarFileName>: <in; separator=", "><endif>
<out:{f | <f> : <grammarFileName>}; separator="\n">
>>
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/depend.stg
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
*   derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```
javaTypeInitMap ::= [
 "int":"0",
 "long":"0",
 "float":"0.0f",
 "double":"0.0",
 "boolean":"false",
 "byte":"0",
 "short":"0",
 "char":"0",
 default:"null" // anything other than a primitive type is an object
]

// args must be <object-model-object>, <fields-resulting-in-STs>
```

```
ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<namedActions.header>
import org.antlr.v4.runtime.atn.*;
import org.antlr.v4.runtime.dfa.DFA;
import org.antlr.v4.runtime.*;
import org.antlr.v4.runtime.misc.*;
import org.antlr.v4.runtime.tree.*;
import java.util.List;
import java.util.Iterator;
import java.util.ArrayList;

<parser>
>>

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<header>
import org.antlr.v4.runtime.tree.ParseTreeListener;

/**
* This interface defines a complete listener for a parse tree produced by
* {@link <file.parserName>}.
*/
public interface <file.grammarName>Listener extends ParseTreeListener {
 <file.listenerNames:{lname |
/**
<if(file.listenerLabelRuleNames.(lname))>
* Enter a parse tree produced by the {@code <lname>\}
* labeled alternative in {@link <file.parserName>#<file.listenerLabelRuleNames.(lname)>\}.
<else>
* Enter a parse tree produced by {@link <file.parserName>#<lname>\}.
<endif>
* @param ctx the parse tree
*/
void enter<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx);
/**
<if(file.listenerLabelRuleNames.(lname))>
* Exit a parse tree produced by the {@code <lname>\}
* labeled alternative in {@link <file.parserName>#<file.listenerLabelRuleNames.(lname)>\}.
<else>
```

```
* Exit a parse tree produced by {@link <file.parserName>#<lname>\}.
<endif>
* @param ctx the parse tree
*/
void exit<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx);}; separator="\n">
}
>>

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<header>

import org.antlr.v4.runtime.ParserRuleContext;
import org.antlr.v4.runtime.tree.ErrorNode;
import org.antlr.v4.runtime.tree.TerminalNode;

/**
* This class provides an empty implementation of {@link <file.grammarName>Listener},
* which can be extended to create a listener which only needs to handle a subset
* of the available methods.
*/
public class <file.grammarName>BaseListener implements <file.grammarName>Listener {
 <file.listenerNames:{lname |
/**
* {@inheritDoc\}
*
* \<p>The default implementation does nothing.\</p>
*/
@Override public void enter<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx) { \}
/**
* {@inheritDoc\}
*
* \<p>The default implementation does nothing.\</p>
*/
@Override public void exit<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx) { \}};
separator="\n">

 /**
  * {@inheritDoc\}
  *
  * \<p>The default implementation does nothing.\</p>
  */
 @Override public void enterEveryRule(ParserRuleContext ctx) { }
 /**
  * {@inheritDoc\}
```

```
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void exitEveryRule(ParserRuleContext ctx) { }
/**
 * {@inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void visitTerminal(TerminalNode node) { }
/**
 * {@inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void visitErrorNode(ErrorNode node) { }
}
>>

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<header>
import org.antlr.v4.runtime.tree.ParseTreeVisitor;

/**
* This interface defines a complete generic visitor for a parse tree produced
* by {@link <file.parserName>}.
*
* @param \<T> The return type of the visit operation. Use {@link Void} for
* operations with no return type.
*/
public interface <file.grammarName>Visitor\<T> extends ParseTreeVisitor\<T> {
 <file.visitorNames:{lname |
/**
<if(file.visitorLabelRuleNames.(lname))>
* Visit a parse tree produced by the {@code <lname>\}
* labeled alternative in {@link <file.parserName>#<file.visitorLabelRuleNames.(lname)>\}.
<else>
* Visit a parse tree produced by {@link <file.parserName>#<lname>\}.
<endif>
* @param ctx the parse tree
* @return the visitor result
*/
T visit<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx);}; separator="\n">
}
```

```
>>

BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<header>
import org.antlr.v4.runtime.tree.AbstractParseTreeVisitor;

/**
* This class provides an empty implementation of {@link <file.grammarName>Visitor},
* which can be extended to create a visitor which only needs to handle a subset
* of the available methods.
*
* @param \<T> The return type of the visit operation. Use {@link Void} for
* operations with no return type.
*/
public class <file.grammarName>BaseVisitor\<T> extends AbstractParseTreeVisitor\<T> implements
<file.grammarName>Visitor\<T> {
 <file.visitorNames:{lname |
/**
* {@inheritDoc\}
*
* \<p>The default implementation returns the result of calling
* {@link #visitChildren\} on {@code ctx\}.\</p>
*/
@Override public T visit<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx) { return
visitChildren(ctx); \}}; separator="\n">
}
>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
// Generated from <grammarFileName; format="java-escape"> by ANTLR <ANTLRVersion>
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
@SuppressWarnings({"all", "warnings", "unchecked", "unused", "cast"})
public class <parser.name> extends <superClass; null="Parser"> {
 static { RuntimeMetaData.checkVersion("<file.ANTLRVersion>", RuntimeMetaData.VERSION); }

 protected static final DFA[] _decisionToDFA;
 protected static final PredictionContextCache _sharedContextCache =
  new PredictionContextCache();
```

```
 <if(parser.tokens)>
 public static final int
  <parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator=", ", wrap, anchor>;
 <endif>
 <if(parser.rules)>
 public static final int
  <parser.rules:{r | RULE_<r.name> = <r.index>}; separator=", ", wrap, anchor>;
 <endif>
 private static String[] makeRuleNames() {
  return new String[] {
   <parser.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
  };
 }
 public static final String[] ruleNames = makeRuleNames();

 <vocabulary(parser.literalNames, parser.symbolicNames)>

 @Override
 public String getGrammarFileName() { return "<parser.grammarFileName; format="java-escape">"; }

 @Override
 public String[] getRuleNames() { return ruleNames; }

 @Override
 public String getSerializedATN() { return _serializedATN; }

 @Override
 public ATN getATN() { return _ATN; }

 <namedActions.members>
 <parser:(ctor)()>
 <funcs; separator="\n">

<if(sempredFuncs)>
 public boolean sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
  switch (ruleIndex) {
   <parser.sempredFuncs.values:{f|
case <f.ruleIndex>:
 return <f.name>_sempred((<f.ctxType>)_localctx, predIndex);}; separator="\n">
  }
  return true;
 }
 <sempredFuncs.values; separator="\n">
<endif>

 <atn>
}
>>
```

```
vocabulary(literalNames, symbolicNames) ::= <<
private static String[] makeLiteralNames() {
 return new String[] {
  <literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
 };
}
private static final String[] _LITERAL_NAMES = makeLiteralNames();
private static String[] makeSymbolicNames() {
 return new String[] {
  <symbolicNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
 };
}
private static final String[] _SYMBOLIC_NAMES = makeSymbolicNames();
public static final Vocabulary VOCABULARY = new VocabularyImpl(_LITERAL_NAMES,
_SYMBOLIC_NAMES);

/**
* @deprecated Use {@link #VOCABULARY} instead.
*/
@Deprecated
public static final String[] tokenNames;
static {
 tokenNames = new String[_SYMBOLIC_NAMES.length];
 for (int i = 0; i \< tokenNames.length; i++) {
  tokenNames[i] = VOCABULARY.getLiteralName(i);
  if (tokenNames[i] == null) {
   tokenNames[i] = VOCABULARY.getSymbolicName(i);
  }

  if (tokenNames[i] == null) {
   tokenNames[i] = "\<INVALID>";
  }
 }
}

@Override
@Deprecated
public String[] getTokenNames() {
 return tokenNames;
}

@Override

public Vocabulary getVocabulary() {
 return VOCABULARY;
}
>>
```

```
dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
@Override
public void action(RuleContext _localctx, int ruleIndex, int actionIndex) {
 switch (ruleIndex) {
 <recog.actionFuncs.values:{f|
case <f.ruleIndex>:
 <f.name>_action((<f.ctxType>)_localctx, actionIndex);
 break;}; separator="\n">
 }
}
<actionFuncs.values; separator="\n">
<endif>
<if(sempredFuncs)>
@Override
public boolean sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
 switch (ruleIndex) {
 <recog.sempredFuncs.values:{f|
case <f.ruleIndex>:
 return <f.name>_sempred((<f.ctxType>)_localctx, predIndex);}; separator="\n">
 }
 return true;
}
<sempredFuncs.values; separator="\n">
<endif>
>>

parser_ctor(p) ::= <<
public <p.name>(TokenStream input) {
 super(input);
 _interp = new ParserATNSimulator(this,_ATN,_decisionToDFA,_sharedContextCache);
}
>>

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<
private void <r.name>_action(<r.ctxType> _localctx, int actionIndex) {
 switch (actionIndex) {
 <actions:{index|
case <index>:
 <actions.(index)>
 break;}; separator="\n">
 }
}
>>
```

```
/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
private boolean <r.name>_sempred(<r.ctxType> _localctx, int predIndex) {
 switch (predIndex) {
 <actions:{index|
case <index>:
 return <actions.(index)>;}; separator="\n">
 }
 return true;
}
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public final <endif><currentRule.ctxType>
<currentRule.name>(<args; separator=",">) throws RecognitionException {
 <currentRule.ctxType> _localctx = new <currentRule.ctxType>(_ctx, getState()<currentRule.args:{a | ,
<a.name>}>);
 enterRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>);
 <namedActions.init>
 <locals; separator="\n">
 try {
<if(currentRule.hasLookaheadBlock)>
  int _alt;
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
 }
 <if(exceptions)>
 <exceptions; separator="\n">
 <else>
 catch (RecognitionException re) {
  _localctx.exception = re;
  _errHandler.reportError(this, re);
  _errHandler.recover(this, re);
 }
 <endif>
 finally {
  <finallyAction>
  exitRule();
```

```
 }
 return _localctx;
}
>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
 namedActions,finallyAction,postamble) ::=
<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public final <endif><currentRule.ctxType>
<currentRule.name>(<args; separator=", ">) throws RecognitionException {
 return <currentRule.name>(0<currentRule.args:{a | , <a.name>}>);
}

private <currentRule.ctxType> <currentRule.name>(int _p<args:{a | , <a>}>) throws RecognitionException {
 ParserRuleContext _parentctx = _ctx;
 int _parentState = getState();
 <currentRule.ctxType> _localctx = new <currentRule.ctxType>(_ctx, _parentState<currentRule.args:{a | ,
<a.name>}>);
 <currentRule.ctxType> _prevctx = _localctx;
 int _startState = <currentRule.startState>;
 enterRecursionRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>, _p);
 <namedActions.init>
 <locals; separator="\n">
 try {
<if(currentRule.hasLookaheadBlock)>
  int _alt;
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
 }
 catch (RecognitionException re) {
  _localctx.exception = re;
  _errHandler.reportError(this, re);
  _errHandler.recover(this, re);
 }
 finally {
  <finallyAction>
  unrollRecursionContexts(_parentctx);
 }
 return _localctx;
}
>>
```

```
CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>_localctx = new <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(_localctx);<endif>
enterOuterAlt(_localctx, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
{
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
}
>>

LL1AltBlock(choice, preamble, alts, error) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = _input.LT(1);<endif>
<preamble; separator="\n">
switch (_input.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
 <alt>
 break;}; separator="\n">
default:
 <error>
}
>>

LL1OptionalBlock(choice, alts, error) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
switch (_input.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
 <alt>
 break;}; separator="\n">
default:
 break;
}
>>

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
<preamble; separator="\n">
if (<expr>) {
 <alts; separator="\n">
}
```

```
<!else if ( !(<followExpr>) ) <error>!>
>>

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
<preamble; separator="\n">
while (<loopExpr>) {
 <alts; separator="\n">
 setState(<choice.loopBackStateNumber>);
 _errHandler.sync(this);
 <iteration>
}
>>

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
setState(<choice.blockStartStateNumber>); <! alt block decision !>
_errHandler.sync(this);
<preamble; separator="\n">
do {
 <alts; separator="\n">
 setState(<choice.stateNumber>); <! loopback/exit decision !>
 _errHandler.sync(this);
 <iteration>
} while ( <loopExpr> );
>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = _input.LT(1);<endif>
<preamble; separator="\n">
switch ( getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx) ) {
<alts:{alt |
case <i>:
 <alt>
 break;}; separator="\n">
}
>>

OptionalBlock(choice, alts, error) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
switch ( getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx) ) {
<alts:{alt |
case <i><if(!choice.ast.greedy)>+1<endif>:
```

```
 <alt>
 break;}; separator="\n">
}
>>

StarBlock(choice, alts, sync, iteration) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
_alt = getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx);
while ( _alt!=<choice.exitAlt> && _alt!=org.antlr.v4.runtime.atn.ATN.INVALID_ALT_NUMBER ) {
 if ( _alt==1<if(!choice.ast.greedy)>+1<endif> ) {
  <iteration>
  <alts> <! should only be one !>
 }
 setState(<choice.loopBackStateNumber>);
 _errHandler.sync(this);
 _alt = getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx);
}
>>

PlusBlock(choice, alts, error) ::= <<
setState(<choice.blockStartStateNumber>); <! alt block decision !>
_errHandler.sync(this);
_alt = 1<if(!choice.ast.greedy)>+1<endif>;
do {
 switch (_alt) {
 <alts:{alt|
case <i><if(!choice.ast.greedy)>+1<endif>:
 <alt>
 break;}; separator="\n">
 default:
  <error>
 }
 setState(<choice.loopBackStateNumber>); <! loopback/exit decision !>
 _errHandler.sync(this);
 _alt = getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx);
} while ( _alt!=<choice.exitAlt> && _alt!=org.antlr.v4.runtime.atn.ATN.INVALID_ALT_NUMBER );
>>

Sync(s) ::= "sync(<s.expecting.name>);"

ThrowNoViableAlt(t) ::= "throw new NoViableAltException(this);"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
>>
```

```
// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShift(s.varName, bits.shift)>})> && ((1L \<\< <offsetShift(s.varName, bits.shift)>) &
(<bits.ttypes:{ttype | (1L \<\< <offsetShift(ttype, bits.shift)>)}; separator=" | ">)) != 0)
%>

isZero ::= [
"0":true,
default:false
]

offsetShift(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(<shiftAmount> - <offset>)<else><shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | <s.varName>==<ttype>}; separator=" || ">
%>

cases(ttypes) ::= <<
<ttypes:{t | case <t>:}; separator="\n">
>>

InvokeRule(r, argExprsChunks) ::= <<
setState(<r.stateNumber>);
<if(r.labels)><r.labels:{l | <labelref(l)> =
}><endif><r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChunks>
);
>>

MatchToken(m) ::= <<
setState(<m.stateNumber>);
<if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>match(<m.name>);
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
setState(<m.stateNumber>);
<if(m.labels)><m.labels:{l | <labelref(l)> = }>_input.LT(1);<endif>
```

```
<capture>
if ( <if(invert)><m.varName> \<= 0 || <else>!<endif>(<expr>) ) {
 <if(m.labels)><m.labels:{l | <labelref(l)> = (Token)}><endif>_errHandler.recoverInline(this);
}
else {
 if ( _input.LA(1)==Token.EOF ) matchedEOF = true;
 _errHandler.reportMatch(this);
 consume();
}
>>

Wildcard(w) ::= <<
setState(<w.stateNumber>);
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>matchWildcard();
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
setState(<p.stateNumber>);
if (!(<chunks>)) throw new FailedPredicateException(this, <p.predicate><if(failChunks)>,
<failChunks><elseif(p.msg)>, <p.msg><endif>);
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
 <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand()  ::= "skip();"
LexerMoreCommand()  ::= "more();"
LexerPopModeCommand() ::= "popMode();"

LexerTypeCommand(arg, grammar)      ::= "_type = <arg>;"
LexerChannelCommand(arg, grammar)   ::= "_channel = <arg>;"
LexerModeCommand(arg, grammar)      ::= "_mode = <arg>;"
LexerPushModeCommand(arg, grammar)  ::= "pushMode(<arg>);"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "_localctx.<a.name>"
```

```
LocalRef(a) ::= "_localctx.<a.name>"
RetValueRef(a) ::= "_localctx.<a.name>"
QRetValueRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>;"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getText():null)"
TokenPropertyRef_type(t) ::= "(<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getType():0)"
TokenPropertyRef_line(t) ::= "(<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getLine():0)"
TokenPropertyRef_pos(t) ::= "(<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getCharPositionInLine():0)"
TokenPropertyRef_channel(t) ::= "(<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getChannel():0)"
TokenPropertyRef_index(t) ::= "(<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getTokenIndex():0)"
TokenPropertyRef_int(t) ::= "(<ctx(t)>.<t.label>!=null?Integer.valueOf(<ctx(t)>.<t.label>.getText()):0)"

RulePropertyRef_start(r) ::= "(<ctx(r)>.<r.label>!=null?(<ctx(r)>.<r.label>.start):null)"
RulePropertyRef_stop(r)  ::= "(<ctx(r)>.<r.label>!=null?(<ctx(r)>.<r.label>.stop):null)"
RulePropertyRef_text(r)  ::=
"(<ctx(r)>.<r.label>!=null?_input.getText(<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop):null)"
RulePropertyRef_ctx(r)  ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r)  ::= "this"

ThisRulePropertyRef_start(r) ::= "_localctx.start"
ThisRulePropertyRef_stop(r)  ::= "_localctx.stop"
ThisRulePropertyRef_text(r)  ::= "_input.getText(_localctx.start, _input.LT(-1))"
ThisRulePropertyRef_ctx(r)  ::= "_localctx"
ThisRulePropertyRef_parser(r)  ::= "this"

NonLocalAttrRef(s)   ::= "((<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>)).<s.name>"
SetNonLocalAttr(s, rhsChunks)   ::=
 "((<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>)).<s.name> = <rhsChunks>;"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.add(<labelref(a.label)>);"

TokenDecl(t) ::= "<TokenLabelType()> <t.name>"
TokenTypeDecl(t) ::= "int <t.name>;"
TokenListDecl(t) ::= "List\<Token> <t.name> = new ArrayList\<Token>()"
RuleContextDecl(r) ::= "<r.ctxName> <r.name>"
RuleContextListDecl(rdecl) ::= "List\<<rdecl.ctxName>> <rdecl.name> = new ArrayList\<<rdecl.ctxName>>()"

ContextTokenGetterDecl(t)      ::=
   "public TerminalNode <t.name>() { return getToken(<parser.name>.<t.name>, 0); }"
ContextTokenListGetterDecl(t)  ::=
```

```
   "public List\<TerminalNode> <t.name>() { return getTokens(<parser.name>.<t.name>); }"
ContextTokenListIndexedGetterDecl(t)  ::= <<
public TerminalNode <t.name>(int i) {
 return getToken(<parser.name>.<t.name>, i);
}
>>
ContextRuleGetterDecl(r)       ::= <<
public <r.ctxName> <r.name>() {
 return getRuleContext(<r.ctxName>.class,0);
}
>>
ContextRuleListGetterDecl(r)   ::= <<
public List\<<r.ctxName>\> <r.name>() {
 return getRuleContexts(<r.ctxName>.class);
}
>>
ContextRuleListIndexedGetterDecl(r)   ::= <<
public <r.ctxName> <r.name>(int i) {
 return getRuleContext(<r.ctxName>.class,i);
}
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
* r becomes rContext.
*/
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "<tokenName>"
ImplicitRuleLabel(ruleName)   ::= "<ruleName>"
ImplicitSetLabel(id)    ::= "_tset<id>"
ListLabelName(label)    ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = _input.LT(1);"
CaptureNextTokenType(d) ::= "<d.varName> = _input.LA(1);"

StructDecl(struct,ctorAttrs,attrs,getters,dispatchMethods,interfaces,extensionMembers)
 ::= <<
public static class <struct.name> extends
<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)> implements
<interfaces; separator=", "><endif> {
 <attrs:{a | public <a>;}; separator="\n">
 <getters:{g | <g>}; separator="\n">
 <if(ctorAttrs)>public <struct.name>(ParserRuleContext parent, int invokingState) { super(parent, invokingState);
}<endif>
 public <struct.name>(ParserRuleContext parent, int invokingState<ctorAttrs:{a | , <a>}>) {
  super(parent, invokingState);
```

```
 <struct.ctorAttrs:{a | this.<a.name> = <a.name>;}; separator="\n">
 }
 @Override public int getRuleIndex() { return RULE_<struct.derivedFromName>; }
<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
 public <struct.name>() { }
 public void copyFrom(<struct.name> ctx) {
  super.copyFrom(ctx);
  <struct.attrs:{a | this.<a.name> = ctx.<a.name>;}; separator="\n">
 }
<endif>
 <dispatchMethods; separator="\n">
 <extensionMembers; separator="\n">
}
>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
public static class <struct.name> extends <currentRule.name; format="cap">Context {
 <attrs:{a | public <a>;}; separator="\n">
 <getters:{g | <g>}; separator="\n">
 public <struct.name>(<currentRule.name; format="cap">Context ctx) { copyFrom(ctx); }
 <dispatchMethods; separator="\n">
}
>>

ListenerDispatchMethod(method) ::= <<
@Override
public void <if(method.isEnter)>enter<else>exit<endif>Rule(ParseTreeListener listener) {
 if ( listener instanceof <parser.grammarName>Listener )
((<parser.grammarName>Listener)listener).<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName;
format="cap">(this);
}
>>

VisitorDispatchMethod(method) ::= <<
@Override
public \<T> T accept(ParseTreeVisitor\<? extends T> visitor) {
 if ( visitor instanceof <parser.grammarName>Visitor ) return ((<parser.grammarName>Visitor\<? extends
T>)visitor).visit<struct.derivedFromName; format="cap">(this);
 else return visitor.visitChildren(this);
}
>>

AttributeDecl(d) ::= "<d.type> <d.name><if(d.initValue)> = <d.initValue><endif>"

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>((<x.ctx.name>)_localctx).<endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr? */
```

```
ctx(actionChunk) ::= "((<actionChunk.ctx.name>)_localctx)"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec)  ::= "precpred(_ctx, <opPrec>)"
recRuleSetReturnAction(src,name)   ::= "$<name>=$<src>.<name>;"
recRuleSetStopToken()                ::= "_ctx.stop = _input.LT(-1);"

recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<
_localctx = new <ctxName>Context(_parentctx, _parentState);
<if(label)>
<if(isListLabel)>
_localctx.<label>.add(_prevctx);
<else>
_localctx.<label> = _prevctx;
<endif>
<endif>
<if(label)>_localctx.<label> = _prevctx;<endif>
pushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
_localctx = new <currentAltLabel; format="cap">Context(new <ruleName; format="cap">Context(_parentctx,
_parentState));
<if(label)>
<if(isListLabel)>
((<currentAltLabel; format="cap">Context)_localctx).<label>.add(_prevctx);
<else>
((<currentAltLabel; format="cap">Context)_localctx).<label> = _prevctx;
<endif>
<endif>
pushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

recRuleReplaceContext(ctxName) ::= <<
_localctx = new <ctxName>Context(_localctx);
_ctx = _localctx;
_prevctx = _localctx;
>>

recRuleSetPrevCtx() ::= <<
if ( _parseListeners!=null ) triggerExitRuleEvent();
_prevctx = _localctx;
>>



LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
<if(lexerFile.genPackage)>
```

```
package <lexerFile.genPackage>;
<endif>
<namedActions.header>
import org.antlr.v4.runtime.Lexer;
import org.antlr.v4.runtime.CharStream;
import org.antlr.v4.runtime.Token;
import org.antlr.v4.runtime.TokenStream;
import org.antlr.v4.runtime.*;
import org.antlr.v4.runtime.atn.*;
import org.antlr.v4.runtime.dfa.DFA;
import org.antlr.v4.runtime.misc.*;

<lexer>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
@SuppressWarnings({"all", "warnings", "unchecked", "unused", "cast"})
public class <lexer.name> extends <superClass; null="Lexer"> {
 static { RuntimeMetaData.checkVersion("<lexerFile.ANTLRVersion>", RuntimeMetaData.VERSION); }

 protected static final DFA[] _decisionToDFA;
 protected static final PredictionContextCache _sharedContextCache =
  new PredictionContextCache();
 <if(lexer.tokens)>
 public static final int
  <lexer.tokens:{k | <k>=<lexer.tokens.(k)>}; separator=", ", wrap, anchor>;
 <endif>
 <if(lexer.channels)>
 public static final int
  <lexer.channels:{c | <c>=<lexer.channels.(c)>}; separator=", ", wrap, anchor>;
 <endif>
 <if(rest(lexer.modes))>
 public static final int
  <rest(lexer.modes):{m | <m>=<i>}; separator=", ", wrap, anchor>;
 <endif>
 public static String[] channelNames = {
  "DEFAULT_TOKEN_CHANNEL", "HIDDEN"<if (lexer.channels)>, <lexer.channels:{c| "<c>"}; separator=", ",
wrap, anchor><endif>
 };

 public static String[] modeNames = {
  <lexer.modes:{m| "<m>"}; separator=", ", wrap, anchor>
 };

 private static String[] makeRuleNames() {
  return new String[] {
   <lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
  };
```

```
}
public static final String[] ruleNames = makeRuleNames();

<vocabulary(lexer.literalNames, lexer.symbolicNames)>

<namedActions.members>

public <lexer.name>(CharStream input) {
 super(input);
 _interp = new LexerATNSimulator(this,_ATN,_decisionToDFA,_sharedContextCache);
}

@Override
public String getGrammarFileName() { return "<lexer.grammarFileName>"; }

@Override
public String[] getRuleNames() { return ruleNames; }

@Override
public String getSerializedATN() { return _serializedATN; }

@Override
public String[] getChannelNames() { return channelNames; }

@Override
public String[] getModeNames() { return modeNames; }

@Override
public ATN getATN() { return _ATN; }

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>
<atn>
}
>>

SerializedATN(model) ::= <<
<if(rest(model.segments))>
<! requires segmented representation !>
private static final int _serializedATNSegments = <length(model.segments)>;
<model.segments:{segment|private static final String _serializedATNSegment<i0> =
 "<segment; wrap={"+<\n><\t>"}>";}; separator="\n">
public static final String _serializedATN = Utils.join(
 new String[] {
  <model.segments:{segment | _serializedATNSegment<i0>}; separator=",\n">
 },
 ""
);
<else>
```

```
<! only one segment, can be inlined !>
public static final String _serializedATN =
 "<model.serialized; wrap={"+<\n><\t>"}>";
<endif>
public static final ATN _ATN =
 new ATNDeserializer().deserialize(_serializedATN.toCharArray());
static {
 _decisionToDFA = new DFA[_ATN.getNumberOfDecisions()];
 for (int i = 0; i \< _ATN.getNumberOfDecisions(); i++) {
  _decisionToDFA[i] = new DFA(_ATN.getDecisionState(i), i);
 }
<! org.antlr.v4.tool.DOTGenerator dot = new org.antlr.v4.tool.DOTGenerator(null);!>
<! System.out.println(dot.getDOT(_ATN.decisionToState.get(0), ruleNames, false));!>
<! System.out.println(dot.getDOT(_ATN.ruleToStartState[2], ruleNames, false));!>
}
>>

/** Using a type to init value map, try to init a type; if not in table
* must be an object, default value is "null".
*/
initValue(typeName) ::= <<
<javaTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".java"
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/codegen/Java/Java.stg
No license file was found, but licenses were detected in source scan.

/*
[The "BSD licence"]
Copyright (c) 2006 Kay Roepke
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
/*
This file contains the actual layout of the messages emitted by ANTLR.
The text itself is coming out of the languages/*stg files, according to the chosen locale.
This file contains the format that mimicks GCC output.
*/

location(file, line, column) ::= "<file>:<line>:<column>:"

message(id, text) ::= "<text> [error <id>]"

report(location, message, type) ::= "<location> <type>: <message>"

wantsSingleLineMessage() ::= "true"

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/messages/formats/gnu.stg
No license file was found, but licenses were detected in source scan.

/*
[The "BSD licence"]
Copyright (c) 2005-2009 Terence Parr
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ANTLRLexer.java
No license file was found, but licenses were detected in source scan.

~ Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/META-INF/maven/org.antlr/antlr4/pom.xml
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
* Copyright (c) 2013 Terence Parr
* Copyright (c) 2013 Sam Harwell
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
*   derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/

// args must be <object-model-object>, <fields-resulting-in-STs>

ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<namedActions.header>
using System;
using System.IO;
using System.Text;
using System.Diagnostics;
using System.Collections.Generic;
using Antlr4.Runtime;
using Antlr4.Runtime.Atn;
using Antlr4.Runtime.Misc;
using Antlr4.Runtime.Tree;
using DFA = Antlr4.Runtime.Dfa.DFA;

<parser>
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>
using Antlr4.Runtime.Misc;
using IParseTreeListener = Antlr4.Runtime.Tree.IParseTreeListener;
using IToken = Antlr4.Runtime.IToken;

/// \<summary>
/// This interface defines a complete listener for a parse tree produced by
/// \<see cref="<csIdentifier.(file.parserName)>"/>.
/// \</summary>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public interface I<file.grammarName>Listener : IParseTreeListener {
 <file.listenerNames:{lname |
/// \<summary>
<if(file.listenerLabelRuleNames.(lname))>
/// Enter a parse tree produced by the \<c><lname>\</c>
```

```
/// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(lname)>"/>.
<else>
/// Enter a parse tree produced by \<see cref="<file.parserName>.<lname>"/>.
<endif>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
void Enter<lname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<lname; format="cap">Context
context);
/// \<summary>
<if(file.listenerLabelRuleNames.(lname))>
/// Exit a parse tree produced by the \<c><lname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(lname)>"/>.
<else>
/// Exit a parse tree produced by \<see cref="<file.parserName>.<lname>"/>.
<endif>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
void Exit<lname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<lname; format="cap">Context
context);}; separator="\n">
}
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>

using Antlr4.Runtime.Misc;
using IErrorNode = Antlr4.Runtime.Tree.IErrorNode;
using ITerminalNode = Antlr4.Runtime.Tree.ITerminalNode;
using IToken = Antlr4.Runtime.IToken;
using ParserRuleContext = Antlr4.Runtime.ParserRuleContext;

/// \<summary>
/// This class provides an empty implementation of \<see cref="I<file.grammarName>Listener"/>,
/// which can be extended to create a listener which only needs to handle a subset
/// of the available methods.
/// \</summary>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.Diagnostics.DebuggerNonUserCode]
[System.CLSCompliant(false)]
public partial class <file.grammarName>BaseListener : I<file.grammarName>Listener {
 <file.listenerNames:{lname |
```

```
/// \<summary>
<if(file.listenerLabelRuleNames.(lname))>
/// Enter a parse tree produced by the \<c><lname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(lname)>"/>.
<else>
/// Enter a parse tree produced by \<see cref="<file.parserName>.<lname>"/>.
<endif>
/// \<para>The default implementation does nothing.\</para>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
public virtual void Enter<lname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<lname;
format="cap">Context context) { \}
/// \<summary>
<if(file.listenerLabelRuleNames.(lname))>
/// Exit a parse tree produced by the \<c><lname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(lname)>"/>.
<else>
/// Exit a parse tree produced by \<see cref="<file.parserName>.<lname>"/>.
<endif>
/// \<para>The default implementation does nothing.\</para>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
public virtual void Exit<lname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<lname;
format="cap">Context context) { \}}; separator="\n">

 /// \<inheritdoc/>
 /// \<remarks>The default implementation does nothing.\</remarks>
 public virtual void EnterEveryRule([NotNull] ParserRuleContext context) { }
 /// \<inheritdoc/>
 /// \<remarks>The default implementation does nothing.\</remarks>
 public virtual void ExitEveryRule([NotNull] ParserRuleContext context) { }
 /// \<inheritdoc/>
 /// \<remarks>The default implementation does nothing.\</remarks>
 public virtual void VisitTerminal([NotNull] ITerminalNode node) { }
 /// \<inheritdoc/>
 /// \<remarks>The default implementation does nothing.\</remarks>
 public virtual void VisitErrorNode([NotNull] IErrorNode node) { }
}
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
```

```
<header>
using Antlr4.Runtime.Misc;
using Antlr4.Runtime.Tree;
using IToken = Antlr4.Runtime.IToken;

/// \<summary>
/// This interface defines a complete generic visitor for a parse tree produced
/// by \<see cref="<csIdentifier.(file.parserName)>"/>.
/// \</summary>
/// \<typeparam name="Result">The return type of the visit operation.\</typeparam>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public interface I<file.grammarName>Visitor\<Result> : IParseTreeVisitor\<Result> {
 <file.visitorNames:{lname |
/// \<summary>
<if(file.visitorLabelRuleNames.(lname))>
/// Visit a parse tree produced by the \<c><lname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.visitorLabelRuleNames.(lname)>"/>.
<else>
/// Visit a parse tree produced by \<see cref="<file.parserName>.<lname>"/>.
<endif>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
/// \<return>The visitor result.\</return>
Result Visit<lname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<lname; format="cap">Context
context);}; separator="\n">
}
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>

BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>
using Antlr4.Runtime.Misc;
using Antlr4.Runtime.Tree;
using IToken = Antlr4.Runtime.IToken;
using ParserRuleContext = Antlr4.Runtime.ParserRuleContext;

/// \<summary>
/// This class provides an empty implementation of \<see cref="I<file.grammarName>Visitor{Result}"/>,
/// which can be extended to create a visitor which only needs to handle a subset
/// of the available methods.
/// \</summary>
```

```
/// \<typeparam name="Result">The return type of the visit operation.\</typeparam>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.Diagnostics.DebuggerNonUserCode]
[System.CLSCompliant(false)]
public partial class <file.grammarName>BaseVisitor\<Result> : AbstractParseTreeVisitor\<Result>,
I<file.grammarName>Visitor\<Result> {
 <file.visitorNames:{lname |
/// \<summary>
<if(file.visitorLabelRuleNames.(lname))>
/// Visit a parse tree produced by the \<c><lname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.visitorLabelRuleNames.(lname)>"/>.
<else>
/// Visit a parse tree produced by \<see cref="<file.parserName>.<lname>"/>.
<endif>
/// \<para>
/// The default implementation returns the result of calling \<see
cref="AbstractParseTreeVisitor{Result\}.VisitChildren(IRuleNode)"/>
/// on \<paramref name="context"/>.
/// \</para>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
/// \<return>The visitor result.\</return>
public virtual Result Visit<lname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<lname;
format="cap">Context context) { return VisitChildren(context); \}}; separator="\n">
}
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
//------------------------------------------------------------------------------
// \<auto-generated>
//     This code was generated by a tool.
//     ANTLR Version: <ANTLRVersion>
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// \</auto-generated>
//------------------------------------------------------------------------------

// Generated from <grammarFileName> by ANTLR <ANTLRVersion>

// Unreachable code detected
#pragma warning disable 0162
// The variable '...' is assigned but its value is never used
#pragma warning disable 0219
// Missing XML comment for publicly visible type or member '...'
```

```
#pragma warning disable 1591
// Ambiguous reference in cref attribute
#pragma warning disable 419

>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public partial class <csIdentifier.(parser.name)> : <superClass; null="Parser"> {
 protected static DFA[] decisionToDFA;
 protected static PredictionContextCache sharedContextCache = new PredictionContextCache();
 <if(parser.tokens)>
 public const int
  <parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator=", ", wrap, anchor>;
 <endif>
 <if(parser.rules)>
 public const int
  <parser.rules:{r | RULE_<r.name> = <r.index>}; separator=", ", wrap, anchor>;
 <endif>
 public static readonly string[] ruleNames = {
  <parser.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
 };

 <vocabulary(parser.literalNames, parser.symbolicNames)>

 public override string GrammarFileName { get { return "<parser.grammarFileName>"; } }

 public override string[] RuleNames { get { return ruleNames; } }

 public override string SerializedAtn { get { return new string(_serializedATN); } }

 static <csIdentifier.(parser.name)>() {
  decisionToDFA = new DFA[_ATN.NumberOfDecisions];
  for (int i = 0; i \< _ATN.NumberOfDecisions; i++) {
   decisionToDFA[i] = new DFA(_ATN.GetDecisionState(i), i);
  }
 }

 <namedActions.members>
 <parser:(ctor)()>
 <funcs; separator="\n">

<if(sempredFuncs)>
```

```
 public override bool Sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
  switch (ruleIndex) {
  <parser.sempredFuncs.values:{f|
case <f.ruleIndex>: return <f.name>_sempred((<f.ctxType>)_localctx, predIndex);}; separator="\n">
  }
  return true;
 }
 <sempredFuncs.values; separator="\n">
<endif>

 <atn>
}
>>

vocabulary(literalNames, symbolicNames) ::= <<
private static readonly string[] _LiteralNames = {
 <literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
};
private static readonly string[] _SymbolicNames = {
 <symbolicNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
};
public static readonly IVocabulary DefaultVocabulary = new Vocabulary(_LiteralNames, _SymbolicNames);

[NotNull]
public override IVocabulary Vocabulary
{
 get
 {
  return DefaultVocabulary;
 }
}
>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
public override void Action(RuleContext _localctx, int ruleIndex, int actionIndex) {
 switch (ruleIndex) {
 <recog.actionFuncs.values:{f|
case <f.ruleIndex> : <f.name>_action(<if(!recog.modes)>(<f.ctxType>)<endif>_localctx, actionIndex); break;};
separator="\n">
 }
}
<actionFuncs.values; separator="\n">
<endif>
<if(sempredFuncs)>
public override bool Sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
 switch (ruleIndex) {
 <recog.sempredFuncs.values:{f|
```

```
case <f.ruleIndex> : return <f.name>_sempred(<if(!recog.modes)>(<f.ctxType>)<endif>_localctx, predIndex);};
separator="\n">
 }
 return true;
}
<sempredFuncs.values; separator="\n">
<endif>
>>

parser_ctor(parser) ::= <<
 public <csIdentifier.(parser.name)>(ITokenStream input) : this(input, Console.Out, Console.Error) { }

 public <csIdentifier.(parser.name)>(ITokenStream input, TextWriter output, TextWriter errorOutput)
 : base(input, output, errorOutput)
{
 Interpreter = new ParserATNSimulator(this, _ATN, decisionToDFA, sharedContextCache);
}
>>

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<
private void <r.name>_action(<r.ctxType> _localctx, int actionIndex) {
 switch (actionIndex) {
 <actions:{index|
case <index>: <actions.(index)> break;}; separator="\n">
 }
}
>>

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
private bool <r.name>_sempred(<r.ctxType> _localctx, int predIndex) {
 switch (predIndex) {
 <actions:{index|
case <index>: return <actions.(index)>;}; separator="\n">
 }
 return true;
}
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<if(ruleCtx)>
```

```
<ruleCtx>
<endif>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

[RuleVersion(<namedActions.version; null="0">)]
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public <endif><currentRule.ctxType>
<csIdentifier.(currentRule.name)>(<args; separator=", ">) {
 <currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, State<currentRule.args:{a | ,
<csIdentifier.(a.name)>}>);
 EnterRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>);
 <namedActions.init>
 <locals; separator="\n">
 try {
<if(currentRule.hasLookaheadBlock)>
  int _alt;
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
 }
 <if(exceptions)>
 <exceptions; separator="\n">
 <else>
 catch (RecognitionException re) {
  _localctx.exception = re;
  ErrorHandler.ReportError(this, re);
  ErrorHandler.Recover(this, re);
 }
 <endif>
 finally {
  <finallyAction>
  ExitRule();
 }
 return _localctx;
}
>>

LeftFactoredRuleFunction(currentRule,args,code,locals,namedActions,finallyAction,postamble) ::=
<<

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>private <endif><currentRule.ctxType>
<csIdentifier.(currentRule.name)>(<args; separator=", ">) {
 <currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, State<currentRule.args:{a | ,
<csIdentifier.(a.name)>}>);
 EnterLeftFactoredRule(_localctx, <currentRule.startState>, RULE_<currentRule.variantOf>);
 <namedActions.init>
 <locals; separator="\n">
 try {
```

```
<if(currentRule.hasLookaheadBlock)>
 int _alt;
<endif>
 <code>
 <postamble; separator="\n">
 <namedActions.after>
 }
 catch (RecognitionException re) {
 _localctx.exception = re;
 ErrorHandler.ReportError(this, re);
 ErrorHandler.Recover(this, re);
 }
 finally {
 <finallyAction>
 ExitRule();
 }
 return _localctx;
}
>>

// This behaves similar to RuleFunction (enterRule is called, and no adjustments
// are made to the parse tree), but since it's still a variant no context class
// needs to be generated.
LeftUnfactoredRuleFunction(currentRule,args,code,locals,namedActions,finallyAction,postamble) ::=
<<

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>private <endif><currentRule.ctxType>
<csIdentifier.(currentRule.name)>(<args; separator=", ">) {
 <currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, State<currentRule.args:{a | ,
<csIdentifier.(a.name)>}>);
 EnterRule(_localctx, <currentRule.startState>, RULE_<currentRule.variantOf>);
 <namedActions.init>
 <locals; separator="\n">
 try {
<if(currentRule.hasLookaheadBlock)>
 int _alt;
<endif>
 <code>
 <postamble; separator="\n">
 <namedActions.after>
 }
 catch (RecognitionException re) {
 _localctx.exception = re;
 ErrorHandler.ReportError(this, re);
 ErrorHandler.Recover(this, re);
 }
 finally {
 <finallyAction>
```

```
 ExitRule();
 }
 return _localctx;
}
>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
 namedActions,finallyAction,postamble) ::=
<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

[RuleVersion(<namedActions.version; null="0">)]
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public <endif><currentRule.ctxType>
<csIdentifier.(currentRule.name)>(<args; separator=", ">) {
 return <csIdentifier.(currentRule.name)>(0<currentRule.args:{a | , <csIdentifier.(a.name)>}>);
}

private <currentRule.ctxType> <csIdentifier.(currentRule.name)>(int _p<args:{a | , <a>}>) {
 ParserRuleContext _parentctx = Context;
 int _parentState = State;
 <currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, _parentState<currentRule.args:{a | ,
<csIdentifier.(a.name)>}>);
 <currentRule.ctxType> _prevctx = _localctx;
 int _startState = <currentRule.startState>;
 EnterRecursionRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>, _p);
 <namedActions.init>
 <locals; separator="\n">
 try {
<if(currentRule.hasLookaheadBlock)>
  int _alt;
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
 }
 catch (RecognitionException re) {
  _localctx.exception = re;
  ErrorHandler.ReportError(this, re);
  ErrorHandler.Recover(this, re);
 }
 finally {
  <finallyAction>
  UnrollRecursionContexts(_parentctx);
 }
 return _localctx;
}
```

```
>>

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>_localctx = new <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(_localctx);<endif>
EnterOuterAlt(_localctx, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
{
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
}
>>

LL1AltBlock(choice, preamble, alts, error) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
<if(choice.label)><labelref(choice.label)> = TokenStream.LT(1);<endif>
<preamble; separator="\n">
switch (TokenStream.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
 <alt>
 break;}; separator="\n">
default:
 <error>
}
>>

LL1OptionalBlock(choice, alts, error) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
switch (TokenStream.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
 <alt>
 break;}; separator="\n">
default:
 break;
}
>>

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
<preamble; separator="\n">
if (<expr>) {
```

```
 <alts; separator="\n">
}
<!else if ( !(<followExpr>) ) <error>!>
>>

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
<preamble; separator="\n">
while (<loopExpr>) {
 <alts; separator="\n">
 State = <choice.loopBackStateNumber>;
 ErrorHandler.Sync(this);
 <iteration>
}
>>

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
State = <choice.blockStartStateNumber>;<! alt block decision !>
ErrorHandler.Sync(this);
<preamble; separator="\n">
do {
 <alts; separator="\n">
 State = <choice.stateNumber>;<! loopback/exit decision !>
 ErrorHandler.Sync(this);
 <iteration>
} while ( <loopExpr> );
>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
<if(choice.label)><labelref(choice.label)> = TokenStream.LT(1);<endif>
<preamble; separator="\n">
switch ( Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context) ) {
<alts:{alt |
case <i>:
 <alt>
 break;}; separator="\n">
}
>>

OptionalBlock(choice, alts, error) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
switch ( Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context) ) {
```

```
<alts:{alt |
case <i><if(!choice.ast.greedy)>+1<endif>:
 <alt>
 break;}; separator="\n">
}
>>

StarBlock(choice, alts, sync, iteration) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
_alt = Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context);
while ( _alt!=<choice.exitAlt> && _alt!=global::Antlr4.Runtime.Atn.ATN.INVALID_ALT_NUMBER ) {
 if ( _alt==1<if(!choice.ast.greedy)>+1<endif> ) {
  <iteration>
  <alts> <! should only be one !>
 }
 State = <choice.loopBackStateNumber>;
 ErrorHandler.Sync(this);
 _alt = Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context);
}
>>

PlusBlock(choice, alts, error) ::= <<
State = <choice.blockStartStateNumber>;<! alt block decision !>
ErrorHandler.Sync(this);
_alt = 1<if(!choice.ast.greedy)>+1<endif>;
do {
 switch (_alt) {
 <alts:{alt|
case <i><if(!choice.ast.greedy)>+1<endif>:
 <alt>
 break;}; separator="\n">
 default:
  <error>
 }
 State = <choice.loopBackStateNumber>;<! loopback/exit decision !>
 ErrorHandler.Sync(this);
 _alt = Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context);
} while ( _alt!=<choice.exitAlt> && _alt!=global::Antlr4.Runtime.Atn.ATN.INVALID_ALT_NUMBER );
>>

Sync(s) ::= "Sync(<s.expecting.name>);"

ThrowNoViableAlt(t) ::= "throw new NoViableAltException(this);"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
```

```
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShift(s.varName, bits.shift)>})> && ((1L \<\< <offsetShift(s.varName, bits.shift)>) &
(<bits.ttypes:{ttype | (1L \<\< <offsetShift(tokenType.(ttype), bits.shift)>)}; separator=" | ">)) != 0)
%>

isZero ::= [
"0":true,
default:false
]

offsetShift(shiftAmount, offset) ::= <%
<if(!isZero.(offset))>(<shiftAmount> - <offset>)<else><shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | <s.varName>==<tokenType.(ttype)>}; separator=" || ">
%>

cases(ttypes) ::= <<
<ttypes:{t | case <tokenType.(t)>:}; separator="\n">
>>

InvokeRule(r, argExprsChunks) ::= <<
State = <r.stateNumber>;
<if(r.labels)><r.labels:{l | <labelref(l)> =
}><endif><csIdentifier.(r.name)>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><arg
ExprsChunks>);
>>

MatchToken(m) ::= <<
State = <m.stateNumber>;
<if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>Match(<tokenType.(m.name)>);
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
```

```
State = <m.stateNumber>;
<if(m.labels)><m.labels:{l | <labelref(l)> = }>TokenStream.LT(1);<endif>
<capture>
if ( <if(invert)><m.varName> \<= 0 || <else>!<endif>(<expr>) ) {
 <if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>ErrorHandler.RecoverInline(this);
}
else {
 ErrorHandler.ReportMatch(this);
   Consume();
}
>>

Wildcard(w) ::= <<
State = <w.stateNumber>;
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>MatchWildcard();
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
State = <p.stateNumber>;
if (!(<chunks>)) throw new FailedPredicateException(this, <p.predicate><if(failChunks)>,
<failChunks><elseif(p.msg)>, <p.msg><endif>);
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
 <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand()  ::= "Skip();"
LexerMoreCommand()  ::= "More();"
LexerPopModeCommand() ::= "PopMode();"

LexerTypeCommand(arg, grammar)      ::= "_type = <tokenType.(arg)>;"
LexerChannelCommand(arg, grammar)   ::= "_channel = <channelName.(arg)>;"
LexerModeCommand(arg, grammar)      ::= "_mode = <modeName.(arg)>;"
LexerPushModeCommand(arg, grammar)  ::= "PushMode(<modeName.(arg)>);"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
```

```
ArgRef(a) ::= "_localctx.<csIdentifier.(a.name)>"
LocalRef(a) ::= "_localctx.<csIdentifier.(a.name)>"
RetValueRef(a) ::= "_localctx.<csIdentifier.(a.name)>"
QRetValueRef(a) ::= "<ctx(a)>.<a.dict>.<csIdentifier.(a.name)>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<csIdentifier.(tokenType.(t.name))>"
LabelRef(t) ::= "<ctx(t)>.<csIdentifier.(t.name)>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(csIdentifier.(t.name))>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<csIdentifier.(s.name)> = <rhsChunks>;"

TokenLabelType() ::= "<file.TokenLabelType; null={IToken}>"
InputSymbolType() ::= "<file.InputSymbolType; null={IToken}>"

TokenPropertyRef_text(t) ::= "(<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Text:null)"
TokenPropertyRef_type(t) ::= "(<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Type:0)"
TokenPropertyRef_line(t) ::= "(<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Line:0)"
TokenPropertyRef_pos(t) ::=
"(<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.CharPositionInLine:0)"
TokenPropertyRef_channel(t) ::=
"(<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Channel:0)"
TokenPropertyRef_index(t) ::=
"(<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.TokenIndex:0)"
TokenPropertyRef_int(t) ::=
"(<ctx(t)>.<tokenType.(t.label)>!=null?int.Parse(<ctx(t)>.<tokenType.(t.label)>.Text):0)"

RulePropertyRef_start(r) ::= "(<ctx(r)>.<r.label>!=null?(<ctx(r)>.<r.label>.Start):null)"
RulePropertyRef_stop(r)  ::= "(<ctx(r)>.<r.label>!=null?(<ctx(r)>.<r.label>.Stop):null)"
RulePropertyRef_text(r)  ::=
"(<ctx(r)>.<r.label>!=null?TokenStream.GetText(<ctx(r)>.<r.label>.Start,<ctx(r)>.<r.label>.Stop):null)"
RulePropertyRef_ctx(r)  ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r)  ::= "this"

ThisRulePropertyRef_start(r) ::= "_localctx.Start"
ThisRulePropertyRef_stop(r)  ::= "_localctx.Stop"
ThisRulePropertyRef_text(r)  ::= "TokenStream.GetText(_localctx.Start, TokenStream.LT(-1))"
ThisRulePropertyRef_ctx(r)  ::= "_localctx"
ThisRulePropertyRef_parser(r)  ::= "this"

NonLocalAttrRef(s)   ::= <%((<s.ruleName;
format="cap">Context)GetInvokingContext(<s.ruleIndex>)).<csIdentifier.(s.name)>%>
SetNonLocalAttr(s, rhsChunks)   ::=
 <%((<s.ruleName; format="cap">Context)GetInvokingContext(<s.ruleIndex>)).<csIdentifier.(s.name)> =
<rhsChunks>;%>

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.Add(<labelref(a.label)>);"

TokenDecl(t) ::= "<TokenLabelType()> <csIdentifier.(tokenType.(t.name))>"
TokenTypeDecl(t) ::= "int <csIdentifier.(tokenType.(t.name))>;"
```

```
TokenListDecl(t) ::= "IList\<IToken> <csIdentifier.(tokenType.(t.name))> = new List\<IToken>()"
RuleContextDecl(r) ::= "<r.ctxName> <csIdentifier.(r.name)>"
RuleContextListDecl(rdecl) ::= "IList\<<rdecl.ctxName>> <csIdentifier.(rdecl.name)> = new
List\<<rdecl.ctxName>>()"

contextGetterCollection(elementType) ::= <%
<elementType>[]
%>

ContextTokenGetterDecl(t)      ::=
   "[System.Diagnostics.DebuggerNonUserCode] public ITerminalNode <csIdentifier.(tokenType.(t.name))>() {
return GetToken(<csIdentifier.(parser.name)>.<csIdentifier.(tokenType.(t.name))>, 0); }"
ContextTokenListGetterDecl(t)  ::= <<
[System.Diagnostics.DebuggerNonUserCode] public <contextGetterCollection("ITerminalNode")>
<csIdentifier.(tokenType.(t.name))>() { return
GetTokens(<csIdentifier.(parser.name)>.<csIdentifier.(tokenType.(t.name))>); }
>>
ContextTokenListIndexedGetterDecl(t)  ::= <<
[System.Diagnostics.DebuggerNonUserCode] public ITerminalNode <csIdentifier.(tokenType.(t.name))>(int i) {
 return GetToken(<csIdentifier.(parser.name)>.<csIdentifier.(tokenType.(t.name))>, i);
}
>>
ContextRuleGetterDecl(r)       ::= <<
[System.Diagnostics.DebuggerNonUserCode] public <r.ctxName> <csIdentifier.(r.name)>() {
 return GetRuleContext\<<r.ctxName>\>(0);
}
>>
ContextRuleListGetterDecl(r)   ::= <<
[System.Diagnostics.DebuggerNonUserCode] public <contextGetterCollection({<r.ctxName>})>
<csIdentifier.(r.name)>() {
 return GetRuleContexts\<<r.ctxName>\>();
}
>>
ContextRuleListIndexedGetterDecl(r)   ::= <<
[System.Diagnostics.DebuggerNonUserCode] public <r.ctxName> <csIdentifier.(r.name)>(int i) {
 return GetRuleContext\<<r.ctxName>\>(i);
}
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
* r becomes rContext.
*/
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenType.(tokenName)>"
ImplicitRuleLabel(ruleName)   ::= "_<ruleName>"
```

```
ImplicitSetLabel(id)    ::= "_tset<id>"
ListLabelName(label)    ::= "_<label>"

CaptureNextToken(d) ::= "<d.varName> = TokenStream.LT(1);"
CaptureNextTokenType(d) ::= "<d.varName> = TokenStream.LA(1);"

StructDecl(struct,ctorAttrs,attrs,getters,dispatchMethods,interfaces,extensionMembers,
        superClass={ParserRuleContext}) ::= <<
public partial class <struct.name> :
<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)>, <interfaces;
separator=", "><endif> {
 <attrs:{a | public <a>;}; separator="\n">
 <getters:{g | <g>}; separator="\n">
 <if(ctorAttrs)>public <struct.name>(ParserRuleContext parent, int invokingState) : base(parent, invokingState) {
}<endif>
 public <struct.name>(ParserRuleContext parent, int invokingState<ctorAttrs:{a | , <a>}>)
  : base(parent, invokingState)
 {
  <struct.ctorAttrs:{a | this.<csIdentifier.(a.name)> = <csIdentifier.(a.name)>;}; separator="\n">
 }
 public override int RuleIndex { get { return RULE_<struct.derivedFromName>; } }
<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
 public <struct.name>() { }
 public virtual void CopyFrom(<struct.name> context) {
  base.CopyFrom(context);
  <struct.attrs:{a | this.<csIdentifier.(a.name)> = context.<csIdentifier.(a.name)>;}; separator="\n">
 }
<endif>
 <dispatchMethods; separator="\n">
 <extensionMembers; separator="\n">
}
>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
public partial class <struct.name> : <currentRule.name; format="cap">Context {
 <attrs:{a | public <a>;}; separator="\n">
 <getters:{g | <g>}; separator="\n">
 public <struct.name>(<currentRule.name; format="cap">Context context) { CopyFrom(context); }
 <dispatchMethods; separator="\n">
}
>>

ListenerDispatchMethod(method) ::= <<
[System.Diagnostics.DebuggerNonUserCode]
public override void <if(method.isEnter)>Enter<else>Exit<endif>Rule(IParseTreeListener listener) {
 I<parser.grammarName>Listener typedListener = listener as I<parser.grammarName>Listener;
 if (typedListener != null) typedListener.<if(method.isEnter)>Enter<else>Exit<endif><struct.derivedFromName;
format="cap">(this);
```

```
}
>>

VisitorDispatchMethod(method) ::= <<
[System.Diagnostics.DebuggerNonUserCode]
public override TResult Accept\<TResult>(IParseTreeVisitor\<TResult> visitor) {
 I<parser.grammarName>Visitor\<TResult> typedVisitor = visitor as I<parser.grammarName>Visitor\<TResult>;
 if (typedVisitor != null) return typedVisitor.Visit<struct.derivedFromName; format="cap">(this);
 else return visitor.VisitChildren(this);
}
>>

AttributeDecl(d) ::= "<d.type> <csIdentifier.(d.name)><if(d.initValue)> = <d.initValue><endif>"

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)><typedContext(x.ctx)>.<endif><csIdentifier.(x.name)>"

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "<typedContext(actionChunk.ctx)>"

// only casts _localctx to the type when the cast isn't redundant (i.e. to a sub-context for a labeled alt)
typedContext(ctx) ::= "<if(ctx.provideCopyFrom)>((<ctx.name>)_localctx)<else>_localctx<endif>"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec)  ::= "Precpred(Context, <opPrec>)"
recRuleSetReturnAction(src,name)   ::= "$<name>=$<src>.<name>;"
recRuleSetStopToken()                 ::= "Context.Stop = TokenStream.LT(-1);"

recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<
_localctx = new <ctxName>Context(_parentctx, _parentState);
<if(label)>
<if(isListLabel)>
_localctx.<label>.Add(_prevctx);
<else>
_localctx.<label> = _prevctx;
<endif>
<endif>
PushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
_localctx = new <currentAltLabel; format="cap">Context(new <ruleName; format="cap">Context(_parentctx,
_parentState));
<if(label)>
<if(isListLabel)>
((<currentAltLabel; format="cap">Context)_localctx).<label>.Add(_prevctx);
<else>
((<currentAltLabel; format="cap">Context)_localctx).<label> = _prevctx;
```

```
<endif>
<endif>
PushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

recRuleReplaceContext(ctxName) ::= <<
_localctx = new <ctxName>Context(_localctx);
Context = _localctx;
_prevctx = _localctx;
>>

recRuleSetPrevCtx() ::= <<
if ( ParseListeners!=null )
 TriggerExitRuleEvent();
_prevctx = _localctx;
>>


LexerFile(file, lexer, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<namedActions.header>
using System;
using System.IO;
using System.Text;
using Antlr4.Runtime;
using Antlr4.Runtime.Atn;
using Antlr4.Runtime.Misc;
using DFA = Antlr4.Runtime.Dfa.DFA;

<lexer>
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public partial class <csIdentifier.(lexer.name)> : <superClass; null="Lexer"> {
 protected static DFA[] decisionToDFA;
 protected static PredictionContextCache sharedContextCache = new PredictionContextCache();
 <if(lexer.tokens)>
 public const int
  <lexer.tokens:{k | <tokenType.(k)>=<lexer.tokens.(k)>}; separator=", ", wrap, anchor>;
 <endif>
```

```
<if(lexer.channels)>
public const int
 <lexer.channels:{k | <csIdentifier.(k)>=<lexer.channels.(k)>}; separator=", ", wrap, anchor>;
<endif>
<if(rest(lexer.modes))>
public const int
 <rest(lexer.modes):{m | <m>=<i>}; separator=", ", wrap, anchor>;
<endif>
public static string[] channelNames = {
 "DEFAULT_TOKEN_CHANNEL", "HIDDEN"<if (lexer.channels)>, <lexer.channels:{c| "<c>"}; separator=", ",
wrap, anchor><endif>
};

public static string[] modeNames = {
 <lexer.modes:{m| "<m>"}; separator=", ", wrap, anchor>
};

public static readonly string[] ruleNames = {
 <lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
};

<namedActions.members>

public <csIdentifier.(lexer.name)>(ICharStream input)
: this(input, Console.Out, Console.Error) { }

public <csIdentifier.(lexer.name)>(ICharStream input, TextWriter output, TextWriter errorOutput)
: base(input, output, errorOutput)
{
 Interpreter = new LexerATNSimulator(this, _ATN, decisionToDFA, sharedContextCache);
}

<vocabulary(lexer.literalNames, lexer.symbolicNames)>

public override string GrammarFileName { get { return "<lexer.grammarFileName>"; } }

public override string[] RuleNames { get { return ruleNames; } }

public override string[] ChannelNames { get { return channelNames; } }

public override string[] ModeNames { get { return modeNames; } }

public override string SerializedAtn { get { return new string(_serializedATN); } }

static <csIdentifier.(lexer.name)>() {
 decisionToDFA = new DFA[_ATN.NumberOfDecisions];
 for (int i = 0; i \< _ATN.NumberOfDecisions; i++) {
  decisionToDFA[i] = new DFA(_ATN.GetDecisionState(i), i);
```

```
 }
 }
 <dumpActions(lexer, "", actionFuncs, sempredFuncs)>
 <atn>
}
>>


SerializedATN(model) ::= <<
private static char[] _serializedATN = {
 <model.serialized; separator=", ", wrap>,
};

public static readonly ATN _ATN =
 new ATNDeserializer().Deserialize(_serializedATN);


>>

initValue(typeName) ::= <<
default(<typeName>)
>>

codeFileExtension() ::= ".cs"

modeName ::= [
 "DEFAULT_MODE" : "DefaultMode",
 default : key
]

channelName ::= [
 "HIDDEN" : "Hidden",
 "DEFAULT_TOKEN_CHANNEL" : "DefaultTokenChannel",
 default : key
]

tokenType ::= [
 "EOF" : "Eof",
 default : key
]

csIdentifier ::= [
 "abstract" : "@abstract",
 "as" : "@as",
 "base" : "@base",
 "bool" : "@bool",
 "break" : "@break",
 "byte" : "@byte",
```

"case" : "@case",
"catch" : "@catch",
"char" : "@char",
"checked" : "@checked",
"class" : "@class",
"const" : "@const",
"continue" : "@continue",
"decimal" : "@decimal",
"default" : "@default",
"delegate" : "@delegate",
"do" : "@do",
"double" : "@double",
"else" : "@else",
"enum" : "@enum",
"event" : "@event",
"explicit" : "@explicit",
"extern" : "@extern",
"false" : "@false",
"finally" : "@finally",
"fixed" : "@fixed",
"float" : "@float",
"for" : "@for",
"foreach" : "@foreach",
"goto" : "@goto",
"if" : "@if",
"implicit" : "@implicit",
"in" : "@in",
"int" : "@int",
"interface" : "@interface",
"internal" : "@internal",
"is" : "@is",
"lock" : "@lock",
"long" : "@long",
"namespace" : "@namespace",
"new" : "@new",
"null" : "@null",
"object" : "@object",
"operator" : "@operator",
"out" : "@out",
"override" : "@override",
"params" : "@params",
"private" : "@private",
"protected" : "@protected",
"public" : "@public",
"readonly" : "@readonly",
"ref" : "@ref",
"return" : "@return",
"sbyte" : "@sbyte",

```
 "sealed" : "@sealed",
 "short" : "@short",
 "sizeof" : "@sizeof",
 "stackalloc" : "@stackalloc",
 "static" : "@static",
 "string" : "@string",
 "struct" : "@struct",
 "switch" : "@switch",
 "this" : "@this",
 "throw" : "@throw",
 "true" : "@true",
 "try" : "@try",
 "typeof" : "@typeof",
 "uint" : "@uint",
 "ulong" : "@ulong",
 "unchecked" : "@unchecked",
 "unsafe" : "@unsafe",
 "ushort" : "@ushort",
 "using" : "@using",
 "virtual" : "@virtual",
 "values" : "@values",
 "void" : "@void",
 "volatile" : "@volatile",
 "while" : "@while",
 default : key
]
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/tool/templates/codegen/CSharp/CSharp.stg
No license file was found, but licenses were detected in source scan.

```
/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
```

```
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

// File      : A3Lexer.g
// Author    : Jim Idle (jimi@temporal-wave.com)
// Copyright : Free BSD - See @header clause below
// Version   : First implemented as part of ANTLR 3.2 this is the self
//             hosting ANTLR 3 Lexer.
//
// Description
// -----------
// This is the definitive lexer grammar for parsing ANTLR V3.x.x grammars. All other
// gramnmars are derived from this grammar via source code control integration (perforce)
// or by the gdiff tool.
//
// This grammar and its associated grmmmars A3Parser.g and A3Walker.g exhibit the following
// traits, which are recommended for all production quality grammars:
//
// 1) They are separate grammars, not composite grammars;
// 2) They implement all supporting methods in a superclass (at least this is recommended
//    for language targets that support inheritence;
// 3) All errors are pushed as far down the parsing chain as possible, which means
//    that the lexer tries to defer error reporting to the parser, and the parser
//    tries to defer error reporting to a semantic phase consisting of a single
//    walk of the AST. The reason for this is that the error messages produced
//    from later phases of the parse will generally have better context and so
//    be more useful to the end user. Consider the message: "Syntax error at 'options'"
//    vs: "You cannot specify two options{} sections in a single grammar file".
// 4) The lexer is 'programmed' to catch common mistakes such as unterminated literals
//    and report them specifically and not just issue confusing lexer mismatch errors.
//

/** Read in an ANTLR grammar and build an AST.  Try not to do
*  any actions, just build the tree.
*
*  The phases are:
*
```

```
*  A3Lexer.g (this file)
*            A3Parser.g
*            A3Verify.g (derived from A3Walker.g)
*  assign.types.g
*  define.g
*  buildnfa.g
*  antlr.print.g (optional)
*  codegen.g
*
*  Terence Parr
*  University of San Francisco
*  2005
*  Jim Idle (this v3 grammar)
*  Temporal Wave LLC
*  2009
*/
lexer grammar ANTLRLexer;

// ============================================================================
// Note that while this grammar does not care about order of constructs
// that don't really matter, such as options before @header etc, it must first
// be parsed by the original v2 parser, before it replaces it. That parser does
// care about order of structures. Hence we are constrained by the v2 parser
// for at least the first bootstrap release that causes this parser to replace
// the v2 version.
// ============================================================================

// -------
// Options
//
// V3 option directives to tell the tool what we are asking of it for this
// grammar.
//
options {

 // Target language is Java, which is the default but being specific
 // here as this grammar is also meant as a good example grammar for
 // for users.
 //
 language      = Java;

 // The super class that this lexer should expect to inherit from, and
 // which contains any and all support routines for the lexer. This is
 // commented out in this baseline (definitive or normative grammar)
 // - see the ANTLR tool implementation for hints on how to use the super
 //   class
 //
 //superclass    = AbstractA3Lexer;
```

```
}

tokens { SEMPRED; TOKEN_REF; RULE_REF; LEXER_CHAR_SET; ARG_ACTION; }

// Include the copyright in this source and also the generated source
//
@lexer::header {
/*
[The "BSD licence"]
Copyright (c) 2005-2009 Terence Parr
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
package org.antlr.v4.parse;
import org.antlr.v4.tool.*;
import org.antlr.v4.runtime.misc.Interval;
}


@members {
 public static final int COMMENTS_CHANNEL = 2;

   public CommonTokenStream tokens; // track stream we push to; need for context info
   public boolean isLexerRule = false;

 public void grammarError(ErrorType etype, org.antlr.runtime.Token token, Object... args) { }
```

```
 /** scan backwards from current point in this.tokens list
  *  looking for the start of the rule or subrule.
  *  Return token or null if for some reason we can't find the start.
  */
 public Token getRuleOrSubruleStartToken() {
     if ( tokens==null ) return null;
  int i = tokens.index();
       int n = tokens.size();
       if ( i>=n ) i = n-1; // seems index == n as we lex
  while ( i>=0 && i<n) {
   int ttype = tokens.get(i).getType();
   if ( ttype == LPAREN || ttype == TOKEN_REF || ttype == RULE_REF ) {
    return tokens.get(i);
   }
   i--;
  }
  return null;
 }
}

// --------
// Comments
//
// ANTLR comments can be multi or single line and we don't care
// which particularly. However we also accept Javadoc style comments
// of the form: /** ... */ and we do take care to distinguish those
// from ordinary multi-line comments
// Note how we guide the lexical PATH because we want to issue a decriptive
// error message in case of a standalone '/' character, which makes no
// sense in ANTLR source code. We alo trap unterminated multi-line comments
//
fragment DOC_COMMENT : ;
COMMENT
@init {

 // Record the start line and offsets as if we need to report an
 // unterminated comment, then we want to show the start of the comment
 // we think is broken, not the end, where people will have to try and work
 // it out themselves.
 //
 int startLine = $line;
 int offset    = getCharPositionInLine();
}
   : // Eat the first character only, then see if we have a comment
     // or something silly.
     //
     '/'  // Comment introducer
```

```
(
    // Single line comment, possibly with embedded src/line directives
    // in a similar style to the C pre-processor, allowing generated
    // code to refer the programmer back to the original source code
    // in case of error.
    //
    '/'
     (
        (' $ANTLR')=> ' $ANTLR' SRC
      | ~(NLCHARS)*
     )

    | // Multi-line comment, which may be a documentation comment
     // if it starts /** (note that we protect against accidentaly
     // recognizing a comment /**/ as a documentation comment
     //
     '*' (
         { input.LA(2) != '/'}?=> '*' { $type = DOC_COMMENT; }
       | { true }?=>  // Required to cover all alts with predicates
         )

         // Should we support embedded multiline comments here?
         //
         (
            // Pick out end of multiline comment and exit the loop
            // if we find it.
            //
            {    !(input.LA(1) == '*' && input.LA(2) == '/') }?

              // Anything else other than the non-greedy match of
              // the comment close sequence
              //
              .
         )*
      (
            // Look for the comment terminator, but if it is accidentally
            // unterminated, then we will hit EOF, which will trigger the
            // epsilon alt and hence we can issue an error message relative
            // to the start of the unterminated multi-line comment
            //
            '*/'

         | // Unterminated comment!
            //
            {
              // ErrorManager.msg(Msg.UNTERMINATED_DOC_COMMENT, startLine, offset, $pos, startLine,
offset, $pos, (Object)null);
            }
```

```
        )

      | // There was nothing that made sense following the opening '/' and so
        // we issue an error regarding the malformed comment
        //
        {
          // TODO: Insert error message relative to comment start
          //
        }
    )
    {
      // We do not wish to pass the comments in to the parser. If you are
      // writing a formatter then you will want to preserve the comments off
      // channel, but could just skip and save token space if not.
      //
      $channel=COMMENTS_CHANNEL;
    }
  ;

ARG_OR_CHARSET
options {k=1;}
  :   {isLexerRule}?=> LEXER_CHAR_SET {$type=LEXER_CHAR_SET;}
  |   {!isLexerRule}?=> ARG_ACTION
      {
      $type=ARG_ACTION;
      // Set the token text to our gathered string minus outer [ ]
      String t = $text;
      t = t.substring(1,t.length()-1);
      setText(t);
      }
  ;

fragment
LEXER_CHAR_SET
 : '['
  ( '\\' ~('\r'|'\n')
  | ~('\r'|'\n'|'\\'|']')
  )*
  ']'
 ;

// ----------
// Argument specs
//
// Certain argument lists, such as those specifying call parameters
// to a rule invocation, or input parameters to a rule specification
// are contained within square brackets. In the lexer we consume them
// all at once and sort them out later in the grammar analysis.
```

```
//
fragment
ARG_ACTION
 : '['
       (
          ARG_ACTION

        | ('"')=>ACTION_STRING_LITERAL

        | ('\'')=>ACTION_CHAR_LITERAL

        | ~('['|']')
      )*

      ']'
 ;

// -------
// Actions
//
// Other than making sure to distinguish between { and } embedded
// within what we have assumed to be literals in the action code, the
// job of the lexer is merely to gather the code within the action
// (delimited by {}) and pass it to the parser as a single token.
// We know that this token will be asked for its text somewhere
// in the upcoming parse, so setting the text here to exclude
// the delimiting {} is no additional overhead.
//
ACTION
 : NESTED_ACTION
  ( '?' {$type = SEMPRED;}
   ( (WSNLCHARS* '=>') => WSNLCHARS* '=>' // v3 gated sempred
    {
    Token t = new CommonToken(input, state.type, state.channel, state.tokenStartCharIndex, getCharIndex()-1);
    t.setLine(state.tokenStartLine);
    t.setText(state.text);
    t.setCharPositionInLine(state.tokenStartCharPositionInLine);
    grammarError(ErrorType.V3_GATED_SEMPRED, t);
    }
   )?
  )?
 ;

// ----------------
// Action structure
//
// Many language targets use {} as block delimiters and so we
// must recursively match {} delimited blocks to balance the
```

```
// braces. Additionally, we must make some assumptions about
// literal string representation in the target language. We assume
// that they are delimited by ' or " and so consume these
// in their own alts so as not to inadvertantly match {}.
// This rule calls itself on matching a {
//
fragment
NESTED_ACTION
@init {

 // Record the start line and offsets as if we need to report an
 // unterminated block, then we want to show the start of the comment
 // we think is broken, not the end, where people will have to try and work
 // it out themselves.
 //
 int startLine = getLine();
 int offset    = getCharPositionInLine();
}

  : // Action and other blocks start with opening {
    //
  '{'
    (
    // And now we can match one of a number of embedded
    // elements within the action until we find a
    // } that balances the opening {. If we do not find
    // the balanced } then we will hit EOF and can issue
    // an error message about the brace that we belive to
    // be mismatched. This won't be foolproof but we will
    // be able to at least report an error against the
    // opening brace that we feel is in error and this will
    // guide the user to the correction as best we can.
    //


      // An embedded {} block
     //
     NESTED_ACTION

     | // What appears to be a literal
       //
       ACTION_CHAR_LITERAL

     | // We have assumed that the target language has C/Java
       // type comments.
       //
       COMMENT
```

```
      | // What appears to be a literal
        //
        ACTION_STRING_LITERAL

    | // What appears to be an escape sequence
      //
      ACTION_ESC

    | // Some other single character that is not
      // handled above
      //
      ~('\\'|''|'\''|'/'|'{'|'}')

    )*

(
    // Correctly balanced closing brace
    //
    '}'

  | // Looks like have an imblanced {} block, report
    // with respect to the opening brace.
    //
    {
       // TODO: Report imbalanced {}
       System.out.println("Block starting  at line " + startLine + " offset " + (offset+1) + " contains imbalanced {} or
is missing a }");
    }
)
 ;


// Keywords
// --------
// keywords used to specify ANTLR v3 grammars. Keywords may not be used as
// labels for rules or in any other context where they would be ambiguous
// with the keyword vs some other identifier
// OPTIONS, TOKENS, and CHANNELS must also consume the opening brace that captures
// their option block, as this is the easiest way to parse it separate
// to an ACTION block, despite it using the same {} delimiters.
//
OPTIONS      : 'options'  WSNLCHARS* '{'  ;
TOKENS_SPEC  : 'tokens'   WSNLCHARS* '{'  ;
CHANNELS     : 'channels' WSNLCHARS* '{'  ;

IMPORT       : 'import'               ;
FRAGMENT     : 'fragment'             ;
LEXER        : 'lexer'                ;
```

```
PARSER      : 'parser'              ;
GRAMMAR     : 'grammar'             ;
TREE_GRAMMAR : 'tree' WSNLCHARS* 'grammar' ;
PROTECTED   : 'protected'           ;
PUBLIC      : 'public'              ;
PRIVATE     : 'private'             ;
RETURNS     : 'returns'             ;
LOCALS      : 'locals'              ;
THROWS      : 'throws'              ;
CATCH       : 'catch'               ;
FINALLY     : 'finally'             ;
MODE        : 'mode'                ;

// -----------
// Punctuation
//
// Character sequences used as separators, delimters, operators, etc
//
COLON       : ':'
            {
            // scan backwards, looking for a RULE_REF or TOKEN_REF.
            // which would indicate the start of a rule definition.
            // If we see a LPAREN, then it's the start of the subrule.
            // this.tokens is the token string we are pushing into, so
            // just loop backwards looking for a rule definition. Then
            // we set isLexerRule.
            Token t = getRuleOrSubruleStartToken();
            if ( t!=null ) {
                if ( t.getType()==RULE_REF ) isLexerRule = false;
                else if ( t.getType()==TOKEN_REF ) isLexerRule = true;
                // else must be subrule; don't alter context
            }
            }
           ;
COLONCOLON  : '::'                  ;
COMMA       : ','                   ;
SEMI        : ';'                   ;
LPAREN      : '('                   ;
RPAREN      : ')'                   ;
RARROW      : '->'                  ;
LT          : '<'                   ;
GT          : '>'                   ;
ASSIGN      : '='                   ;
QUESTION    : '?'                   ;
SYNPRED     : '=>'
    {
     Token t = new CommonToken(input, state.type, state.channel,
                        state.tokenStartCharIndex, getCharIndex()-1);
```

```
    t.setLine(state.tokenStartLine);
    t.setText(state.text);
    t.setCharPositionInLine(state.tokenStartCharPositionInLine);
    grammarError(ErrorType.V3_SYNPRED, t);
            $channel=HIDDEN;
    }
        ;
STAR        : '*'                ;
PLUS        : '+'                ;
PLUS_ASSIGN  : '+='                ;
OR          : '|'                ;
DOLLAR      : '$'                ;
DOT         : '.'                ; // can be WILDCARD or DOT in qid or imported rule ref
RANGE       : '..'               ;
AT          : '@'                ;
POUND       : '#'                ;
NOT         : '~'                ;
RBRACE      : '}'                ;

/** Allow unicode rule/token names */
ID   : a=NameStartChar NameChar*
    {
    if ( Grammar.isTokenName($a.text) ) $type = TOKEN_REF;
    else $type = RULE_REF;
    }
  ;

fragment
NameChar    :   NameStartChar
        |   '0'..'9'
        |   '_'
        |   '\u00B7'
        |   '\u0300'..'\u036F'
        |   '\u203F'..'\u2040'
        ;

fragment
NameStartChar
        :   'A'..'Z' | 'a'..'z'
        |   '\u00C0'..'\u00D6'
        |   '\u00D8'..'\u00F6'
        |   '\u00F8'..'\u02FF'
        |   '\u0370'..'\u037D'
        |   '\u037F'..'\u1FFF'
        |   '\u200C'..'\u200D'
        |   '\u2070'..'\u218F'
        |   '\u2C00'..'\u2FEF'
        |   '\u3001'..'\uD7FF'
```

```
        |   '\uF900'..'\uFDCF'
        |   '\uFDF0'..'\uFEFE'
        |   '\uFF00'..'\uFFFD'
        ; // ignores | ['\u10000-'\uEFFFF] ;

// ----------------------------
// Literals embedded in actions
//
// Note that we have made the assumption that the language used within
// actions uses the fairly standard " and ' delimiters for literals and
// that within these literals, characters are escaped using the \ character.
// There are some languages which do not conform to this in all cases, such
// as by using /string/ and so on. We will have to deal with such cases if
// if they come up in targets.
//

// Within actions, or other structures that are not part of the ANTLR
// syntax, we may encounter literal characters. Within these, we do
// not want to inadvertantly match things like '}' and so we eat them
// specifically. While this rule is called CHAR it allows for the fact that
// some languages may use/allow ' as the string delimiter.
//
fragment
ACTION_CHAR_LITERAL
 : '\'' (('\\')=>ACTION_ESC | ~'\'' )* '\''
 ;

// Within actions, or other structures that are not part of the ANTLR
// syntax, we may encounter literal strings. Within these, we do
// not want to inadvertantly match things like '}' and so we eat them
// specifically.
//
fragment
ACTION_STRING_LITERAL
 : '"' (('\\')=>ACTION_ESC | ~'"')* '"'
 ;

// Within literal strings and characters that are not part of the ANTLR
// syntax, we must allow for escaped character sequences so that we do not
// inadvertantly recognize the end of a string or character when the terminating
// delimiter has been esacped.
//
fragment
ACTION_ESC
 : '\\' .
 ;

// -------
```

```
// Integer
//
// Obviously (I hope) match an aribtrary long sequence of digits.
//
INT : ('0'..'9')+
  ;

// -----------
// Source spec
//
// A fragment rule for picking up information about an origrinating
// file from which the grammar we are parsing has been generated. This allows
// ANTLR to report errors against the originating file and not the generated
// file.
//
fragment
SRC : 'src' WSCHARS+ file=ACTION_STRING_LITERAL WSCHARS+ line=INT
    {
      // TODO: Add target specific code to change the source file name and current line number
      //
    }
  ;

// --------------
// Literal string
//
// ANTLR makes no disticintion between a single character literal and a
// multi-character string. All literals are single quote delimited and
// may contain unicode escape sequences of the form \uxxxx or \u{xxxxxx},
// where x is a valid hexadecimal number.
STRING_LITERAL
  :  '\'' ( ( ESC_SEQ | ~('\\'|'\''|'\r'|'\n') ) )*
    (    '\''
    |    // Unterminated string literal
         {
         Token t = new CommonToken(input, state.type, state.channel, state.tokenStartCharIndex, getCharIndex()-1);
         t.setLine(state.tokenStartLine);
         t.setText(state.text);
         t.setCharPositionInLine(state.tokenStartCharPositionInLine);
         grammarError(ErrorType.UNTERMINATED_STRING_LITERAL, t);
         }
    )
  ;

// A valid hex digit specification
//
fragment
HEX_DIGIT : ('0'..'9'|'a'..'f'|'A'..'F') ;
```

```
// Any kind of escaped character that we can embed within ANTLR
// literal strings.
//
fragment
ESC_SEQ
  : '\\'
    (
        // The standard escaped character set such as tab, newline, etc...
      'b'|'t'|'n'|'f'|'r'|'\"'|'\\'

      | // A Java style Unicode escape sequence
        UNICODE_ESC

        | // A Swift/Hack style Unicode escape sequence
          UNICODE_EXTENDED_ESC

      | // An illegal escape seqeunce
        ~('b'|'t'|'n'|'f'|'r'|'\"'|'\\'|'u') // \x for any invalid x (make sure to match char here)
        {
            Token t = new CommonToken(input, state.type, state.channel, getCharIndex()-2, getCharIndex()-1);
            t.setText(t.getText());
            t.setLine(input.getLine());
            t.setCharPositionInLine(input.getCharPositionInLine()-2);
            grammarError(ErrorType.INVALID_ESCAPE_SEQUENCE, t, input.substring(getCharIndex()-
2,getCharIndex()-1));
        }
    )
  ;

fragment
UNICODE_ESC
@init {

 // Flag to tell us whether we have a valid number of
 // hex digits in the escape sequence
 //
 int hCount = 0;
}
  :  'u' // Leadin for unicode escape sequence

     // We now require 4 hex digits. Note though
     // that we accept any number of characters
     // and issue an error if we do not get 4. We cannot
     // use an inifinite count such as + because this
     // might consume too many, so we lay out the lexical
     // options and issue an error at the invalid paths.
     //
```

```
  (
    (
      HEX_DIGIT  { hCount++; }
       (
          HEX_DIGIT  { hCount++; }
        (
            HEX_DIGIT  { hCount++; }
            (
               // Four valid hex digits, we are good
               //
               HEX_DIGIT { hCount++; }

             | // Three valid digits
            )

          | // Two valid digits
        )

       | // One valid digit
    )
     )
    | // No valid hex digits at all
  )

  // Now check the digit count and issue an error if we need to
  //
  {
   if (hCount < 4) {
  Interval badRange = Interval.of(getCharIndex()-2-hCount, getCharIndex());
  String lastChar = input.substring(badRange.b, badRange.b);
  if ( lastChar.codePointAt(0)=='\' ) {
   badRange.b--;
  }
  String bad = input.substring(badRange.a, badRange.b);
  Token t = new CommonToken(input, state.type, state.channel, badRange.a, badRange.b);
  t.setLine(input.getLine());
  t.setCharPositionInLine(input.getCharPositionInLine()-hCount-2);
  grammarError(ErrorType.INVALID_ESCAPE_SEQUENCE, t, bad);
   }
  }
 ;

fragment
UNICODE_EXTENDED_ESC
  :  'u{' // Leadin for unicode extended escape sequence

     HEX_DIGIT+ // One or more hexadecimal digits
```

```
      '}' // Leadout for unicode extended escape sequence

      // Now check the digit count and issue an error if we need to
      {
         int numDigits = getCharIndex()-state.tokenStartCharIndex-6;
         if (numDigits > 6) {
            Token t = new CommonToken(input, state.type, state.channel, state.tokenStartCharIndex, getCharIndex()-
1);
            t.setText(t.getText());
            t.setLine(input.getLine());
            t.setCharPositionInLine(input.getCharPositionInLine()-numDigits);
            grammarError(ErrorType.INVALID_ESCAPE_SEQUENCE, t,
input.substring(state.tokenStartCharIndex,getCharIndex()-1));
  }
      }
  ;

// ----------
// Whitespace
//
// Characters and character constructs that are of no import
// to the parser and are used to make the grammar easier to read
// for humans.
//
WS
  : (
     ' '
     | '\t'
     | '\r'
     | '\n'
     | '\f'
    )+
    {$channel=HIDDEN;}
  ;

// A fragment rule for use in recognizing end of line in
// rules like COMMENT.
//
fragment
NLCHARS
  : '\n' | '\r'
  ;

// A fragment rule for recognizing traditional whitespace
// characters within lexer rules.
//
fragment
WSCHARS
```

```
  : ' ' | '\t' | '\f'
  ;

// A fragment rule for recognizing both traditional whitespace and
// end of line markers, when we don't care to distinguish but don't
// want any action code going on.
//
fragment
WSNLCHARS
  : ' ' | '\t' | '\f' | '\n' | '\r'
  ;

// This rule allows ANTLR 4 to parse grammars using the UTF-8 encoding with a
// byte order mark. Since this Unicode character doesn't appear as a token
// anywhere else in the grammar, we can simply skip all instances of it without
// problem. This rule will not break usage of \uFEFF inside a LEXER_CHAR_SET or
// STRING_LITERAL.
UnicodeBOM
  :   '\uFEFF' {skip();}
  ;

// ------------------
// Illegal Character
//
// This is an illegal character trap which is always the last rule in the
// lexer specification. It matches a single character of any value and being
// the last rule in the file will match when no other rule knows what to do
// about the character. It is reported as an error but is not passed on to the
// parser. This means that the parser to deal with the grammar file anyway
// but we will not try to analyse or code generate from a file with lexical
// errors.
//
ERRCHAR
  : .
    {
      Token t = new CommonToken(input, state.type, state.channel, state.tokenStartCharIndex, getCharIndex()-1);
      t.setLine(state.tokenStartLine);
      t.setText(state.text);
      t.setCharPositionInLine(state.tokenStartCharPositionInLine);
      String msg = getTokenErrorDisplay(t) + " came as a complete surprise to me";
      grammarError(ErrorType.SYNTAX_ERROR, t, msg);
      state.syntaxErrors++;
      skip();
    }
  ;

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ANTLRLexer.g
```

No license file was found, but licenses were detected in source scan.

```
/*
* [The "BSD license"]
*  Copyright (c) 2012-2016 Terence Parr
*  Copyright (c) 2012-2016 Sam Harwell
*  All rights reserved.
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*  1. Redistributions of source code must retain the above copyright
*     notice, this list of conditions and the following disclaimer.
*  2. Redistributions in binary form must reproduce the above copyright
*     notice, this list of conditions and the following disclaimer in the
*     documentation and/or other materials provided with the distribution.
*  3. The name of the author may not be used to endorse or promote products
*     derived from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
*  IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
*  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
*  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
*  NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
*  THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

lexer grammar ActionSplitter;

options { filter=true; }

@header {
package org.antlr.v4.parse;
import org.antlr.v4.tool.*;
import org.antlr.v4.tool.ast.*;
}

@members {
ActionSplitterListener delegate;

public ActionSplitter(CharStream input, ActionSplitterListener delegate) {
  this(input, new RecognizerSharedState());
  this.delegate = delegate;
```

```
}

/** force filtering (and return tokens). triggers all above actions. */
public List<Token> getActionTokens() {
  List<Token> chunks = new ArrayList<Token>();
  Token t = nextToken();
  while ( t.getType()!=Token.EOF ) {
     chunks.add(t);
     t = nextToken();
  }
  return chunks;
}

private boolean isIDStartChar(int c) {
 return c == '_' || Character.isLetter(c);
}
}

// ignore comments right away

COMMENT
  :   '/*' ( options {greedy=false;} : . )* '*/' {delegate.text($text);}
  ;

LINE_COMMENT
  : '//' ~('\n'|'\r')* '\r'? '\n' {delegate.text($text);}
  ;

SET_NONLOCAL_ATTR
 : '$' x=ID '::' y=ID WS? '=' expr=ATTR_VALUE_EXPR ';'
  {
  delegate.setNonLocalAttr($text, $x, $y, $expr);
  }
 ;

NONLOCAL_ATTR
 : '$' x=ID '::' y=ID {delegate.nonLocalAttr($text, $x, $y);}
 ;

QUALIFIED_ATTR
 : '$' x=ID '.' y=ID {input.LA(1)!='('}? {delegate.qualifiedAttr($text, $x, $y);}
 ;

SET_ATTR
 : '$' x=ID WS? '=' expr=ATTR_VALUE_EXPR ';'
  {
  delegate.setAttr($text, $x, $expr);
  }
```

```
 ;

ATTR
 : '$' x=ID {delegate.attr($text, $x);}
 ;

// Anything else is just random text
TEXT
@init {StringBuilder buf = new StringBuilder();}
@after {delegate.text(buf.toString());}
 : ( c=~('\\'| '$') {buf.append((char)$c);}
  | '\\$' {buf.append('$');}
  | '\\' c=~('$') {buf.append('\\').append((char)$c);}
  | {!isIDStartChar(input.LA(2))}? => '$' {buf.append('$');}
  )+
 ;

fragment
ID  : ('a'..'z'|'A'..'Z'|'_') ('a'..'z'|'A'..'Z'|'0'..'9'|'_')*
   ;

/** Don't allow an = as first char to prevent $x == 3; kind of stuff. */
fragment
ATTR_VALUE_EXPR
 : ~'=' (~';')*
 ;

fragment
WS : (' '|'\t'|'\n'|'\r')+
 ;
```

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/ActionSplitter.g
No license file was found, but licenses were detected in source scan.

```
/*
[The "BSD license"]
Copyright (c) 2011 Terence Parr
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
```

documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1300428068_1649144266.79/0/antlr4-4-9-2-sources-jar/org/antlr/v4/parse/GrammarTreeVisitor.java

1.198 okio 3.0.0

1.198.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/-Base64.kt
No license file was found, but licenses were detected in source scan.

/*

```
* Copyright (C) 2020 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/HashingSink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/ForwardingFileSystem.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/internal/-Path.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/FileMetadata.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/Path.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/ExperimentalFileSystem.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/FileSystem.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/NioSystemFileSystem.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/Path.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/JvmSystemFileSystem.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/HashingSource.kt
No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2021 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
```

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/JvmFileHandle.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/internal/FixedLengthSource.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/internal/-FileSystem.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/internal/ResourceFileSystem.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/FileHandle.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/FileSystem.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2020 Square, Inc. and others.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/CipherSink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/CipherSource.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/ForwardingSink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/RealBufferedSink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/Buffer.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/Segment.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/AsyncTimeout.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/DeflaterSink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/ForwardingSource.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/GzipSink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/JvmOkio.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/SegmentPool.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/GzipSource.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/SegmentPool.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/Timeout.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/BufferedSink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/RealBufferedSource.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/BufferedSource.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/InflaterSource.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/ByteString.kt

* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/-Util.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/-CommonPlatform.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/internal/-ByteString.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/-JvmPlatform.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/internal/-Utf8.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/-DeprecatedUpgrade.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/-DeprecatedUtf8.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/PeekSource.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/-DeprecatedOkio.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/Throttler.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2017 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/Utf8.kt
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/internal/ZipEntry.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/ZipFileSystem.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/internal/zip.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/Options.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/HashingSink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/Pipe.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/HashingSource.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/RealBufferedSource.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/Sink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/RealBufferedSink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/Buffer.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/BufferedSource.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/Okio.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/internal/-Buffer.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/BufferedSink.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/Timeout.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/Source.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/internal/-SegmentedByteString.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/internal/-RealBufferedSource.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/internal/-RealBufferedSink.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 Square Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/ByteString.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2015 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/commonMain/okio/SegmentedByteString.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/ForwardingTimeout.kt
* /opt/cola/permits/1519136853_1671710551.8664162/0/okio-3-0-0-sources-2-jar/jvmMain/okio/SegmentedByteString.kt

1.199 elasticsearch-kubernetes-cloud-plugin 5.12.1

1.199.1 Available under license :

No license file was found, but licenses were detected in source scan.

/**
* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* Encodes and decodes to and from Base64 notation.
* Homepage: http://iharder.net/base64.
*
* Example:
*
* <code>String encoded = Base64.encode(myByteArray);</code>
*
* <code>byte[] myByteArray = Base64.decode(encoded);</code>

```
*
* <p>The <tt>options</tt> parameter, which appears in a few places, is used to pass
* several pieces of information to the encoder. In the "higher level" methods such as
* encodeBytes( bytes, options ) the options parameter can be used to indicate such
* things as first gzipping the bytes before encoding them, not inserting linefeeds,
* and encoding using the URL-safe and Ordered dialects.</p>
*
* <p>Note, according to <a href="http://www.faqs.org/rfcs/rfc3548.html">RFC3548</a>,
* Section 2.1, implementations should not add line feeds unless explicitly told
* to do so. I've got Base64 set to this behavior now, although earlier versions
* broke lines by default.</p>
*
* <p>The constants defined in Base64 can be OR-ed together to combine options, so you
* might make a call like this:</p>
*
* <code>String encoded = Base64.encodeBytes( mybytes, Base64.GZIP | Base64.DO_BREAK_LINES );</code>
* <p>to compress the data before encoding it and then making the output have newline characters.</p>
* <p>Also...</p>
* <code>String encoded = Base64.encodeBytes( crazyString.getBytes() );</code>
*
*
*
* <p>
* Change Log:
* </p>
* <ul>
*  <li>v2.3.7 - Fixed subtle bug when base 64 input stream contained the
*   value 01111111, which is an invalid base 64 character but should not
*   throw an ArrayIndexOutOfBoundsException either. Led to discovery of
*   mishandling (or potential for better handling) of other bad input
*   characters. You should now get an IOException if you try decoding
*   something that has bad characters in it.</li>
*  <li>v2.3.6 - Fixed bug when breaking lines and the final byte of the encoded
*   string ended in the last column; the buffer was not properly shrunk and
*   contained an extra (null) byte that made it into the string.</li>
*  <li>v2.3.5 - Fixed bug in #encodeFromFile where estimated buffer size
*   was wrong for files of size 31, 34, and 37 bytes.</li>
*  <li>v2.3.4 - Fixed bug when working with gzipped streams whereby flushing
*   the Base64.OutputStream closed the Base64 encoding (by padding with equals
*   signs) too soon. Also added an option to suppress the automatic decoding
*   of gzipped streams. Also added experimental support for specifying a
*   class loader when using the
*   #decodeToObject(java.lang.String, int, java.lang.ClassLoader)
*   method.</li>
*  <li>v2.3.3 - Changed default char encoding to US-ASCII which reduces the internal Java
*   footprint with its CharEncoders and so forth. Fixed some javadocs that were
*   inconsistent. Removed imports and specified things like java.io.IOException
*   explicitly inline.</li>
```

```
* <li>v2.3.2 - Reduced memory footprint! Finally refined the "guessing" of how big the
*  final encoded data will be so that the code doesn't have to create two output
*  arrays: an oversized initial one and then a final, exact-sized one. Big win
*  when using the {@link #encodeBytesToBytes(byte[])} family of methods (and not
*  using the gzip options which uses a different mechanism with streams and stuff).</li>
* <li>v2.3.1 - Added {@link #encodeBytesToBytes(byte[], int, int, int)} and some
*  similar helper methods to be more efficient with memory by not returning a
*  String but just a byte array.</li>
* <li>v2.3 - <strong>This is not a drop-in replacement!</strong> This is two years of comments
*  and bug fixes queued up and finally executed. Thanks to everyone who sent
*  me stuff, and I'm sorry I wasn't able to distribute your fixes to everyone else.
*  Much bad coding was cleaned up including throwing exceptions where necessary
*  instead of returning null values or something similar. Here are some changes
*  that may affect you:
*  <ul>
*   <li><em>Does not break lines, by default.</em> This is to keep in compliance with
*    <a href="http://www.faqs.org/rfcs/rfc3548.html">RFC3548</a>.</li>
*   <li><em>Throws exceptions instead of returning null values.</em> Because some operations
*    (especially those that may permit the GZIP option) use IO streams, there
*    is a possiblity of an java.io.IOException being thrown. After some discussion and
*    thought, I've changed the behavior of the methods to throw java.io.IOExceptions
*    rather than return null if ever there's an error. I think this is more
*    appropriate, though it will require some changes to your code. Sorry,
*    it should have been done this way to begin with.</li>
*   <li><em>Removed all references to System.out, System.err, and the like.</em>
*    Shame on me. All I can say is sorry they were ever there.</li>
*   <li><em>Throws NullPointerExceptions and IllegalArgumentExceptions</em> as needed
*    such as when passed arrays are null or offsets are invalid.</li>
*   <li>Cleaned up as much javadoc as I could to avoid any javadoc warnings.
*    This was especially annoying before for people who were thorough in their
*    own projects and then had gobs of javadoc warnings on this file.</li>
*  </ul>
* <li>v2.2.1 - Fixed bug using URL_SAFE and ORDERED encodings. Fixed bug
*  when using very small files (~&lt; 40 bytes).</li>
* <li>v2.2 - Added some helper methods for encoding/decoding directly from
*  one file to the next. Also added a main() method to support command line
*  encoding/decoding from one file to the next. Also added these Base64 dialects:
*  <ol>
*  <li>The default is RFC3548 format.</li>
*  <li>Calling Base64.setFormat(Base64.BASE64_FORMAT.URLSAFE_FORMAT) generates
*  URL and file name friendly format as described in Section 4 of RFC3548.
*  http://www.faqs.org/rfcs/rfc3548.html</li>
*  <li>Calling Base64.setFormat(Base64.BASE64_FORMAT.ORDERED_FORMAT) generates
*  URL and file name friendly format that preserves lexical ordering as described
*  in http://www.faqs.org/qa/rfcc-1940.html</li>
*  </ol>
*  Special thanks to Jim Kellerman at <a href="http://www.powerset.com/">http://www.powerset.com/</a>
*  for contributing the new Base64 dialects.
```

```
*  </li>
*
*  <li>v2.1 - Cleaned up javadoc comments and unused variables and methods. Added
*   some convenience methods for reading and writing to and from files.</li>
*  <li>v2.0.2 - Now specifies UTF-8 encoding in places where the code fails on systems
*   with other encodings (like EBCDIC).</li>
*  <li>v2.0.1 - Fixed an error when decoding a single byte, that is, when the
*   encoded data was a single byte.</li>
*  <li>v2.0 - I got rid of methods that used booleans to set options.
*   Now everything is more consolidated and cleaner. The code now detects
*   when data that's being decoded is gzip-compressed and will decompress it
*   automatically. Generally things are cleaner. You'll probably have to
*   change some method calls that you were making to support the new
*   options format (<tt>int</tt>s that you "OR" together).</li>
*  <li>v1.5.1 - Fixed bug when decompressing and decoding to a
*   byte[] using <tt>decode( String s, boolean gzipCompressed )</tt>.
*   Added the ability to "suspend" encoding in the Output Stream so
*   you can turn on and off the encoding if you need to embed base64
*   data in an otherwise "normal" stream (like an XML file).</li>
*  <li>v1.5 - Output stream pases on flush() command but doesn't do anything itself.
*     This helps when using GZIP streams.
*     Added the ability to GZip-compress objects before encoding them.</li>
*  <li>v1.4 - Added helper methods to read/write files.</li>
*  <li>v1.3.6 - Fixed OutputStream.flush() so that 'position' is reset.</li>
*  <li>v1.3.5 - Added flag to turn on and off line breaks. Fixed bug in input stream
*     where last buffer being read, if not completely full, was not returned.</li>
*  <li>v1.3.4 - Fixed when "improperly padded stream" error was thrown at the wrong time.</li>
*  <li>v1.3.3 - Fixed I/O streams which were totally messed up.</li>
* </ul>
*
* <p>
* I am placing this code in the Public Domain. Do with it as you will.
* This software comes with no guarantees or warranties but with
* plenty of well-wishing instead!
* Please visit <a href="http://iharder.net/base64">http://iharder.net/base64</a>
* periodically to check for updates or to contribute improvements.
* </p>
*
* @author Robert Harder
* @author rob@iharder.net
* @version 2.3.7
*/
```

Found in path(s):
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/Base64.java

No license file was found, but licenses were detected in source scan.

<!--

  Copyright (C) 2015 Red Hat, Inc.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

        http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.

-->

Found in path(s):
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-client/pom.xml
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/lib/FilenameUtils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/lib/FileSystem.java
No license file was found, but licenses were detected in source scan.

/**

* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1SchedulingAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Ttyable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Recreateable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1PolicyAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/WatcherWebSocketListener.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/ReplaceValueStream.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/ReplicaSetOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/DiscoveryAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ResourceNotFoundException.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1ApiextensionAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/run/RunConfig.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/PodOperationUtil.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/CreateListFromLoadable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Operation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/SSLUtils.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Tailable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/DeploymentRollingUpdater.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ContainerResource.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ScalableResource.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/PodControllerOperationContext.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/BatchAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V2beta2AutoscalingAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/base/ResourceDefinitionContext.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1NetworkAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/leaderelection/LeaderCallbacks.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1beta1PolicyAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/KubernetesListOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/StatusEditable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/FilterWatchListMultiDeletable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/FlowControlAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/base/PatchType.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/core/v1/BindingOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/AbstractWatchManager.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1AdmissionRegistrationAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/MixedOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1DiscoveryAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1CertificatesAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/BytesLimitTerminateTimeTailPrettyLoggable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/StatusPatchable.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/InputStreamPumper.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/KubeConfigUtils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Copiable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ApiextensionsAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/RollingOperationContext.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/IOHelpers.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/AdmissionRegistrationAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/okhttp/OkHttpClientBuilderImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/OptionalDependencyWrapper.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/http/WebSocket.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Scaleable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/NamespaceListVisitFromServerGetDeleteRecreateWaitApplicable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/KubernetesListNonNamespaceOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/StatusUpdatable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1CertificatesAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/DeleteAndCreateHelper.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/cache/SharedProcessor.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V2beta2AutoscalingAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1CertificatesAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/NetworkAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/okhttp/OkHttpClientImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1FlowControlAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/Version.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/PodResource.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ExtensionsAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1CertificatesAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/ExponentialBackoffIntervalCalculator.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ExecListenable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/AppsAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1ApiextensionsAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/URLFromNodePortImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/PodMetricOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/extensions/v1beta1/DeploymentRollingUpdater.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/HttpClientUtils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/PatchUtils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/base/OperationSupport.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1NetworkAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1AdmissionRegistrationAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/VersionUsageUtils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/EditReplacePatchDeletable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ExecListener.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/AdmissionRegistrationAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/URLUtils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Deletable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Versionable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/core/v1/PodOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Containerable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/cache/Indexer.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/http/TlsVersion.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ListVisitFromServerGetDeleteRecreateWaitApplicable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/WatchConnectionManager.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/http/HttpClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1beta1DiscoveryAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/CachedSingleThreadScheduler.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Prettyable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/FileSelector.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/patchmixins/ObjectMetaMixIn.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/ImpersonatorInterceptor.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/okhttp/OkHttpWebSocketImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ErrorChannelable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/CertificateSigningRequestResource.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/CustomResource.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1AuthorizationAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ApplicableAnd.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/PortForwardable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/http/HttpHeaders.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Secretable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/PropagationPolicyConfigurable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/BaseKubernetesClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/WritableOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/FromServerGettable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/serialization/BeanPropertyWriterDelegate.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/Utils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1ApiextensionAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1beta1CertificatesAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/AutoscalingAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/HttpClientAware.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/run/RunOperations.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/leaderelection/resourcelock/Lock.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/RequestConfigurable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/HasMetadataOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V2beta1AutoscalingAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/KubernetesClientException.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/osgi/ManagedKubernetesClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Gettable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/EventingAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/okhttp/OkHttpRequestImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/okhttp/OkHttpClientFactory.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1AuthorizationAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/KubernetesClientTimeoutException.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V2beta2AutoscalingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/PortForwarderWebsocket.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ExecWatch.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1DiscoveryAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/leaderelection/LeaderElectionConfig.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/Adapters.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Editable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/SerializationUtils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/ResourceCompare.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1beta1BatchAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/LogWatch.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/PolicyAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Terminateable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/TtyExecErrorable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/IpAddressMatcher.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/SchedulingAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ServiceResource.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1BatchAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/APIGroupNotAvailableException.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/PodOperationContext.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/RawCustomResourceOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/leaderelection/resourcelock/LeaderElectionRecord.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/PKCS1Util.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/GenericKubernetesClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/EditReplacePatchable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/TtyExecable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/CertUtils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Listenable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/StorageAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/DryRunable.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Watchable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/Client.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1SchedulingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/MetricAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Createable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1AdmissionRegistrationAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/core/v1/ReplicationControllerRollingUpdater.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/http/BasicBuilder.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Outputable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1ApiextensionsAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/base/BaseOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/SchedulingAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ServiceToURLProvider.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/RbacAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/FlowControlAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/leaderelection/package-info.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Filterable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Timeable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/BackwardsCompatibilityInterceptor.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V2beta1AutoscalingAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/VersionInfo.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/ReflectUtils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1ApiextensionsAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/http/HttpRequest.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ParameterNamespaceListVisitFromServerGetDeleteRecreateWaitApplicable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/KubernetesListOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/GracePeriodConfigurable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/VisitFromServerGetWatchDeleteRecreateWaitApplicable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Nameable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V2beta1AutoscalingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/URLFromServiceUtil.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/WatchAndWaitable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/SimpleClientContext.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Readiable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/NonNamespaceOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/KubernetesListMixedOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/CascadingDeletable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1AdmissionRegistrationAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Timestampable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1NetworkAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/FlowControlAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/NetworkAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1PolicyAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1AuthorizationAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/WatchListDeletable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/KubernetesVersionPriority.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/RecreateApplicable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/cache/Store.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/DeploymentOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/Watch.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/VisitFromServerWritable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/base/FilterNestedImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1EventingAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/leaderelection/resourcelock/ConfigMapLock.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ParameterMixedOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/AppsAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/MultiDeleteable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Applicable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/SharedInformerFactory.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/core/v1/ReplicationControllerOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/DiscoveryAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/cache/ProcessorListener.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/Config.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/CreateOrReplaceable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/PortForward.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/DiscoveryAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/RbacAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/NamespacedKubernetesClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1APIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/SharedInformerEventListener.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/RequestConfig.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/CreateOrReplaceHelper.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/InOutCreateable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1AutoscalingAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/OAuthTokenProvider.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/PodStatusUtil.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Execable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/http/HttpResponse.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/NamespaceVisitFromServerGetWatchDeleteRecreateWaitApplicableListImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ApiextensionsAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/Handlers.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/base/OperationContext.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/NamespaceVisitOperationContext.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/NetworkAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/uploadable/PodUpload.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/OperationInfo.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ResourceHandler.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/KubernetesVersionFactory.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Waitable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/leaderelection/LeaderElector.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1AuthorizationAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/CascadingEditReplacePatchDeletable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Lockable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/CertificatesAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1FlowControlAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-

jar/io/fabric8/kubernetes/client/dsl/RecreateFromServerGettable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/BytesLimitable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/NamespaceVisitFromServerGetWatchDeleteRecreateWaitApplicableImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1DiscoveryAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Deniable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/RbacAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/impl/DefaultSharedIndexInformer.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/FilterNested.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/readiness/Readiness.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/URLFromIngressImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/StorageAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/leaderelection/LeaderElectorBuilder.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Requirable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/MetricOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Parameterizable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1AdmissionRegistrationAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/URLFromClusterIPImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/leaderelection/resourcelock/LockException.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/EventingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/APIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/base/CreateOnlyResourceOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1beta1FlowControlAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/SerialExecutor.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/MetricAPIGroupClient.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1AdmissionRegistrationAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/FromServerable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1BatchAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/TtyExecInputOutputErrorable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/CertificatesAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/cache/SyncableStore.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1BatchAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/NodeMetricOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/LogWatchCallback.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/StatusReplaceable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ImageEditReplacePatchable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/DefaultKubernetesClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/SchedulingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1EventingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1CertificatesAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/BatchAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/FunctionCallable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1beta1EventingAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1PolicyAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/ResyncRunnable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/NamespacedInOutCreateable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/cache/ProcessorStore.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/TtyExecErrorChannelable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1EventingAPIGroupClient.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/uploadable/PodUploadWebSocketListener.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Rollable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/run/RunConfigUtil.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1SchedulingAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Resource.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/BaseClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1PolicyAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/PolicyAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/TerminateTimeTailPrettyLoggable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/extended/leaderelection/resourcelock/LeaseLock.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/PortForwarder.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/StatefulSetOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Timeoutable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/TokenRefreshInterceptor.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/core/v1/ServiceOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1PolicyAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/AuthorizationAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ExtensionsAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/Watcher.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/ResourceEventHandler.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/CertificatesAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ConfigAware.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/batch/v1/JobOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1AutoscalingAPIGroupExtensionAdapter.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/http/Interceptor.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/RecreateCreateable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/cache/Lister.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Listable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/AutoAdaptableKubernetesClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/GenericKubernetesResourceOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/ListerWatcher.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ListVisitFromServerWritable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Cascading.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/WatcherToggle.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/ApiVersionUtil.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1NetworkAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/ReplaceDeletable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/VersionWatchAndWaitable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/http/WebSocketHandshakeException.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/KubernetesResourceUtil.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/SharedIndexInformer.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/serializationmixins/ReplicationControllerMixIn.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Typeable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/BatchAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/CustomResourceList.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/http/HttpStatusMessage.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/RollableScalableResourceOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/TtyExecOutputErrorable.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/WithRequestCallable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/TimeoutImageEditReplacePatchable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1NetworkAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1AuthorizationAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/Serialization.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/extensions/v1beta1/DeploymentOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1NetworkAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/URLFromEnvVarsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/AnyNamespaceable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Loadable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1BatchAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/CopyOrReadable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/MetricAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/FilterWatchListDeletable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Triggerable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/AuthorizationAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/LocalPortForward.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/core/v1/ComponentStatusOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/OpenIDConnectionUtils.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ResourceHandlerImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1APIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/AutoscalingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Informable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/KubernetesClient.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/AuthorizationAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/StatefulSetRollingUpdater.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/AutoscalingAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/extensions/v1beta1/ReplicaSetOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/base/PatchContext.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1EventingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/PrettyLoggable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/RequestConfigHolder.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/CreateFromServerGettable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/WatchHTTPManager.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1SchedulingAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1SchedulingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/extensions/v1beta1/ReplicaSetRollingUpdater.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1beta1SchedulingAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/internal/serializationmixins/ObjectMetaMixIn.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Loggable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/TimestampBytesLimitTerminateTimeTailPrettyLoggable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/CreateOrDeleteable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1DiscoveryAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/ExecWebSocketListener.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Uploadable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Readable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Namespaceable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/GetListFromLoadable.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Errorable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/base/HasMetadataOperation.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1APIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/certificates/v1/CertificateSigningRequestOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Patchable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1EventingAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/base/CustomResourceDefinitionContext.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/TimeTailPrettyLoggable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1BatchAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Inputable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/ReplicaSetRollingUpdater.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ExtensionsAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/AdmissionRegistrationAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/serialization/SettableBeanPropertyDelegate.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/WatcherException.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/RollableScalableResource.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/V1DiscoveryAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/utils/serialization/UnmatchedFieldTypeModule.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1ApiextensionsAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/CreateOnlyResourceOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Approvable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ExtensionAdapterSupport.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Replaceable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/TailPrettyLoggable.java

* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/RollingUpdater.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/DefaultOperationInfo.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/AppsAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/cache/Reflector.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/batch/v1beta1/CronJobOperationsImpl.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/cache/Cache.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/PolicyAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/StorageAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/EventingAPIGroupClient.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/informers/SharedInformer.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1AutoscalingAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/Evictable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/dsl/NamespaceVisitFromServerGetWatchDeleteRecreateWaitApplicable.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ApiextensionsAPIGroupExtensionAdapter.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/V1beta1AuthorizationAPIGroupDSL.java
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/io/fabric8/kubernetes/client/ClientContext.java
No license file was found, but licenses were detected in source scan.

#
Copyright (C) 2015 Red Hat, Inc.
#
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
#
http://www.apache.org/licenses/LICENSE-2.0
#
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

#

io.fabric8.kubernetes.client.URLFromEnvVarsImpl
io.fabric8.kubernetes.client.URLFromIngressImpl
io.fabric8.kubernetes.client.URLFromNodePortImpl
io.fabric8.kubernetes.client.URLFromClusterIPImpl

Found in path(s):
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/META-INF/services/io.fabric8.kubernetes.client.ServiceToURLProvider
No license file was found, but licenses were detected in source scan.

#
Copyright (C) 2015 Red Hat, Inc.
#
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
#
http://www.apache.org/licenses/LICENSE-2.0
#
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
#

io.fabric8.kubernetes.client.AppsAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.AdmissionRegistrationAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1AdmissionRegistrationAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1AdmissionRegistrationAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.AutoscalingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.ApiextensionsAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.AuthorizationAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1AutoscalingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V2beta1AutoscalingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V2beta2AutoscalingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.BatchAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1BatchAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1BatchAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.ExtensionsAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.EventingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1EventingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1EventingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.FlowControlAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1FlowControlAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.MetricAPIGroupExtensionAdapter

io.fabric8.kubernetes.client.NetworkAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.PolicyAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1PolicyAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1PolicyAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.RbacAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.SchedulingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1SchedulingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1SchedulingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.StorageAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1APIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1ApiextensionsAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1ApiextensionsAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1AuthorizationAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1AuthorizationAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1NetworkAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1NetworkAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.DiscoveryAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1DiscoveryAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1DiscoveryAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.CertificatesAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1CertificatesAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1CertificatesAPIGroupExtensionAdapter

Found in path(s):
* /opt/cola/permits/1340815648_1654861194.3826184/0/kubernetes-client-5-12-1-sources-jar/META-INF/services/io.fabric8.kubernetes.client.ExtensionAdapter

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries.To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

©2022 Cisco Systems, Inc. All rights reserved.